НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Кваліфікаційна наукова
праця на правах рукопису

Міщук Олександра Сергіївна

УДК 004.67+004.89+004.942

ДИСЕРТАЦІЯ
НЕЙРОПОДІБНІ МЕТОДИ ТА ЗАСОБИ ПРОГНОЗУВАННЯ ПАРАМЕТРІВ ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ

05.13.23 — системи та засоби штучного інтелекту

Подається на здобуття наукового ступеня кандидата технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Міщук О.С.

Науковий керівник — Ткаченко Роман Олексійович, доктор технічних наук, професор

Львів — 2020

АНОТАЦІЯ

Міщук Олександра Сергіївна. Нейроподібні методи та засоби прогнозування параметрів забруднення атмосферного повітря. — Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук (доктора філософії) за спеціальністю 05.13.23 — системи та засоби штучного інтелекту. Національний університет "Львівська політехніка" Міністерства освіти і науки України, Львів, 2020.

Дисертаційне дослідження присвячено підвищенню ефективності прогнозування параметрів забруднення атмосферного повітря, як завдання моніторингу довкілля, на основі розроблених, удосконалених та розвинених нейроподібних методів та засобів прогнозування параметрів забруднення атмосферного повітря в умовах пропусків у даних.

У *вступі* обґрунтовано актуальність теми дисертаційного дослідження; описано зв'язок роботи з науковими програмами, планами, темами; сформульовано мету та завдання дисертаційної роботи; представлено методи дослідження та визначено наукову новизну під час розв'язання поставлених завдань. Також відображено практичне значення одержаних результатів дослідження; презентовано списки опублікованих праць за тематикою дисертаційної роботи та конференцій, на котрих було апробовано основні результати дисертаційної роботи.

У *першому розділі* проаналізовано систему моніторингу навколишнього середовища та досліджено її завдання, серед яких розрізняють організацію спостережень, оцінювання виміряних даних та прогнозування стану довкілля. Тому, відповідно до завдань моніторингу довкілля, та враховуючи види і масштаби об'єктів спостереження, виконано аналізування низки наукових українських та закордонних робіт у визначеній сфері та обгрунтовано дослідження локального екологічного моніторингу атмосферного повітря.

Досліджено, що основними об’єктами викидів забруднюючих речовин у атмосферне повітря є тваринництво та сільське господарство, енергетичні та промислові підприємства, всі види транспорту, аварії та інші. Низький рівень оснащення вже працюючих джерел викидів пилогазоочисним обладнанням призводить до шкідливих викидів у атмосферу таких параметрів забруднення як діоксид сірки, діоксид азоту, оксид вуглецю, метан, аміак, пил, неметанові летючі органічні сполуки, зважені суспендовані частинки та інші. Визначено, що для контролю за станом навколишнього середовища та для вчасного прийняття природохоронних дій потрібно виконувати прогнозування параметрів забруднення атмосферного повітря, в тому числі в умовах пропусків у даних.

Проаналізувавши методи заповнення пропущених параметрів забруднення, обґрунтовано дослідження методу середнього значення та моделей на основі регресійного аналізу, серед яких виділено метод опорних векторів, лінійну регресію зі стохастичним градієнтним спуском, адаптивний бустинг, дерево рішень та нейроподібну структуру моделі послідовних геометричних перетворень.

Аналіз моделей прогнозування параметрів забруднення у даних моніторингу атмосферного повітря показав, що моделі повинні відображати зміни, які відбуваються в повітрі під впливом людської діяльності; своєчасно забезпечувати підсистеми моделювання якісною інформацією про стан довкілля; та включати в себе ретроспективний аналіз існуючих прогнозів. Крім того, досліджено евристичні, аналітичні та статистичні підходи до прогнозування параметрів забруднення атмосферного повітря та виконано аналіз нейромережевих методів прогнозування: багатошарового перцептрона, нейронної мережі радіальних базисних функцій, нейронної мережі узагальненої регресії, рекурентної нейронної мережі, та всіх інших, котрі здатні апроксимувати нелінійні функції. Також проаналізовано нейроподібні структури моделі послідовних геометричних перетворень (НС МПГП), що не використовувалися раніше для розв’язку задач прогнозування в умовах частково пропущених параметрів забруднення атмосферного повітря.

Таким чином, досліджено, що перспективним є розробка нових методів прогнозування параметрів забруднення атмосферного повітря, в тому числі в умовах пропусків у даних моніторингу довкілля на основі НС МПГП з метою підвищення точності застосування на мобільних та вбудованих пристроях.

Заповнення пропущених концентрацій параметрів забруднення у даних моніторингу атмосферного повітря за допомогою штучних нейронних мереж чи нейроподібних структур відбувається за рахунок виконання їх прогнозування. Тому у *другому розділі* запропоновано розроблення універсального методу підвищення точності прогнозування вихідних значень за рахунок формування додаткових вхідних ознак векторів концентрацій параметрів забруднення атмосферного повітря. Описано два методи розширення вхідних вибірок даних: розроблений метод на основі виділення компактних множ точок та удосконалений метод на основі обернено-пропорційних квадратичних функцій.

Виконано порівняльну оцінку ефективності регресійних методів заповнення пропусків у даних моніторингу забруднення атмосферного повітря та методу середнього значення. Доведено, що найефективнішим методом заповнення пропусків у даних екологічного моніторингу є метод на основі НС МПГП, оскільки результує точнішими результатами, ніж інші досліджувані методи, що описані вище. Встановлено, що середня похибка заповнення пропущених концентрацій параметрів забруднення атмосферного повітря на основі НС МПГП становить 21,9 %.

Досліджено, що використання розробленого методу формування додаткових вхідних ознак векторів даних за допомогою попереднього виділення компактних множин забезпечує підвищення точності заповнення пропущених параметрів забруднення повітряного середовища. Доведено, що середня відносна похибка заповнення пропущених концентрацій параметрів забруднення атмосферного повітря за допомогою НС МПГП зменшилась на 8,5 %. Також встановлено, що розроблений метод спрацьовує не для всіх досліджених методів. Наприклад, під час застосування опорних векторів із розширеними вибірками даних точність заповнення пропущених атрибутів зменшується на 1,5 %.

Також, у другому розділі описано процедуру удосконалення методу функційного розширення входів Йох-Хан Пао шляхом застосування обернено-пропорційних квадратичних функцій, що забезпечило зниження похибок в режимі застосування на невідомих при навчанні даних на 4,3 %.

У *третьому розділі* розроблено та досліджено метод короткотермінового прогнозування часових послідовностей показників параметрів забруднення повітряного середовища за допомогою комітету лінійної та нелінійної нейроподібних структур для збільшення горизонту прогнозування тренду забруднення атмосферного повітря за рахунок часткового коректування окремо додатних і від'ємних відхилень від точних значень. Корекція відхилень (похибок) виконується за умови, що похибка прогнозування нелінійною нейроподібною структурою є меншою за похибку прогнозування лінійною нейроподібною структурою. Використання методу на основі корекції похибки забезпечило зменшення похибки прогнозування чадного газу розробленим методом відносно НС МПГП на 15 % та збільшило горизонт прогнозування на два дні.

Також, розвинено метод нейромережевої ідентифікації коефіцієнтів поліномів за рахунок побудови матриці коефіцієнтів лінійних поліномів, створеної шляхом їх ідентифікації за результатами навчання лінійної нейроподібної структури моделі послідовних геометричних перетворень. Експериментально доведено, що розвинутий метод забезпечив зменшення затрат пам'яті під час прогнозування параметрів забруднення атмосферного повітря в 12,75 разів.

У *четвертому розділі* реалізовано нейромережеві та нейроподібні моделі та методи прогнозування параметрів забруднення атмосферного повітря в тому числі в умовах пропусків у даних (описаних у попередніх розділах). Розроблено програмний засіб, що реалізовано двома мовами – Java та Python. У розробленому Python-фреймворку виконано налаштування параметрів, навчання та застосування нейромережевих та нейроподібних моделей та методів прогнозування/заповнення пропусків у даних моніторингу атмосферного повітря.

Для реалізації розроблених методів підвищення точності прогнозування параметрів забруднення повітряного середовища в умовах пропусків у даних використано мову Java. Програмний засіб з розробленими методами підвищення точності прогнозування параметрів забруднення атмосферного повітря на мобільних та вбудованих пристроях включає в себе наступні модулі: власне набір розроблених, удосконалених та розвинутих методів; модуль взаємодії двох програмних проектів; модуль обробки даних моніторингу атмосферного повітря; модуль з класами, що містить методи для роботи з таблицями даних; модуль реалізації методів прогнозування та модуль обрахунку точності та швидкодії реалізованих нейромережевих та нейроподібних методів. Також у Java-фреймворку розроблений користувацький інтерфейс для можливості прогнозування на мобільних пристроях.

Набір розроблених методів підвищення точності та швидкості прогнозування параметрів забруднення атмосферного повітря в умовах пропусків у даних моніторингу повітряного середовища включає в себе такі розроблені в Java-фреймворку методи: метод введення додаткових вхідних ознак векторів даних на основі попереднього виділення компактних множин точок; метод функційного розширення входів Йох-Хан Пао на основі раціональних дробів; метод короткотермінового прогнозування параметрів забруднення повітряного середовища на основі корекції похибки комітетом нейроподібних структур різних типів та метод побудови апроксимаційних лінійних поліномів шляхом ідентифікації їх коефіцієнтів за результатами навчання нейроподібної структури моделі послідовних геометричних перетворень.

На основі обчислених похибок прогнозування в умовах пропущених концентрацій параметрів забруднення атмосферного повітря було визначено найефективніші методи. Експериментально доведено, що методи на основі нейроподібних структур моделі послідовних геометричних перетворень показують точніші результати прогнозування параметрів забруднення атмосферного повітря та використовують менше оперативної пам’яті, ніж інші досліджені моделі та методи.

Ключові слова: моніторинг довкілля, параметри забруднення, часові ряди, нейроподібні структури моделі послідовних геометричних перетворень, пропущені дані, кластеризація, прогнозування тренду.

Список публікацій аспірантки:

Статті у наукових фахових виданнях України:

1. Міщук О. С. Нейронна мережа з комбінованою апроксимацією поверхні відгуку / О.С. Міщук, П. Б. Вітинський // Наукові вісті КПІ: міжнародний науково-технічний журнал. — 2018. — № 2. — С. 18-24.
2. Міщук О.С. Методи оброблення та заповнення пропущених параметрів у даних екологічного моніторингу / О. С. Міщук, Р. О. Ткаченко // Науковий вісник НЛТУ України. — 2019. — №29(6). — С. 119-122. doi: 10.15421/40290623
3. Міщук О. С. Багатокрокове прогнозування тренду показників забруднення атмосферного повітря // Науковий вісник НЛТУ України. — 2019. — №29(8). — С. 142-146.
4. Mishchuk O. Development of the method of forecasting the atmospheric air pollution parameters based on error correction by neural-like structures of the model of successive geometric transformations // Technology Audit and Production Reserves. — 2019. — № 6/2(50). — P. 22–26.

Стаття у науковому періодичному виданні іншої держави:

5. Mishchuk O. The Accelerated Method of Filling Gaps in Data Using a Linear SGTM Neural-Like Structure / O. Mishchuk, R. Tkachenko, V. Pohrebennyk // International Journal of Science and Engineering Investigations (IJSEI). — 2019. — №8(91). — P. 154-159.

Матеріали конференцій у наукових серійних закордонних виданнях, що включено до міжнародних наукометричних баз:

6. Mishchuk O. Missing Data Imputation Through SGTM Neural-Like Structure for Environmental Monitoring Tasks / O. Mishchuk, R. Tkachenko, I. Izonin // Advances in Computer Science for Engineering and Education II. — International

Conference on Computer Science, Engineering and Education Applications, ICCSEEA 2019. — Springer. — Vol. 938. — P. 142-151. doi: 10.1007/978-3-030-16621-2_13 *(Scopus)*

7. Izonin I. SGD-Based Wiener Polynomial Approximation for Missing Data Recovery in Air Pollution Monitoring Dataset / I. Izonin, M. Greguš, R. Tkachenko, M. Logoyda, O. Mishchuk, Y. Kynash // Advances in Computational Intelligence. — 15th International Work-Conference on Artificial Neural Networks, IWANN 2019. — Vol. 11506. — P. 781-793. doi: 10.1007/978-3-030-20521-8_64 *(Scopus)*
8. Tkachenko R. A Non-Iterative Neural-Like Framework for Missing Data Imputation / R. Tkachenko, O. Mishchuk, I. Izonin, N. Kryvinska, R. Stoliarchuk // Procedia Computer Science. — The 14th International Conference on Future Networks and Communications, FNC 2019. — Elsevier. — Vol. 155. — P. 319-326. doi: 10.1016/j.procs.2019.08.046 *(Scopus)*
9. Izonin I. Recovery of Incomplete IoT Sensed Data using High-Performance Extended-Input Neural-Like Structure / I. Izonin, R. Tkachenko, N. Kryvinska, K. Zub, O. Mishchk, T. Lisovych // Procedia Computer Science. — International Workshop on Digitalization and Servitization within Factory-Free Economy, D&SwFFE-2019. — Elsevier. — Vol. 160. — P. 521-526. *(Scopus)*
10. Mishchuk O. One-step Prediction of Air Pollution Control Parameters using Neural-Like Structure Based on Geometric Data Transformations / O. Mishchuk, R. Tkachenko // Electronics and Information Technologies (ELIT-2019) : Proceedings of the XIth International Scientific and Practical Conference, Lviv, 16-18 September 2019. — Lviv. — P. 192-197. *(Scopus)*

Тези доповіді конференцій:

11. Міщук О.С. Нейроподібні структури моделі геометричних перетворень з комбінованою апроксимацією поверхні відгуку // Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту (ISDMCI-2018) : матеріали XIV-ї міжнародної наукової конференції, Залізний Порт, 21-27 травня 2018. — Херсон. — С. 87-89.

12. Міщук О. С. Нелінійне розширення входів нейронної структури моделі послідовних геометричних перетворень // Обчислювальні методи і системи перетворення інформації (ОМІСПІ-2018) : збірник праць V-ї науково-технічної конференції, Львів, 4-5 жовтня 2018. — Львів. — С. 126-129.
13. Міщук О.С. Відновлення пропусків у даних моніторингу забруднення повітря за допомогою нейронної структури моделі послідовних геометричних перетворень // Комп'ютерне моделювання та оптимізація складних систем (КМОСС-2018) : матеріали IV-ї міжнародної науково-технічної конференції, Дніпро, 1-2 листопада 2018. — Дніпро. — С. 260-261.
14. Mishchuk O. Expansion of Neural-like structures inputs using combined approximation / O. Mishchuk, R. Tkachenko // Computer and Information Systems and Technologies (CSITIC-2019) : Proceedings of the IIIrd International Scientific Conference, Kharkiv, 23-24 April 2019. — Kharkiv. — P. 29-32.
15. Міщук О. С. Прогнозування параметрів забруднення атмосферного повітря за допомогою лінійних нейроподібних структур // Побудова інформаційного суспільства: ресурси і технології: матеріали XVIII-ї міжнародної науково-практичної конференції, Київ, 19-20 вересня 2019. — Київ. — С. 275-278.
16. Mishchuk O. Neural network method of forecasting the air pollution trend by carbon monoxide // Information Technologies and Automation (ITA-2019) : Proceedings of the XIIth International Scientific Conference, Odesa, 17-18 October, 2019. — Odesa. — P. 101-102.
17. Міщук О.С. Підвищення точності прогнозування параметрів забруднення повітря // Комп'ютерне моделювання та оптимізація складних систем (КМОСС-2019) : матеріали V-ї міжнародної науково-технічної конференції, Дніпро, 6-8 листопада 2019. — Дніпро. — С.129-130.

ANNOTATION

Mishchuk Oleksandra. Neural-like methods and tools to forecast the parameters of atmospheric air pollution. Qualified scientific work on the rights of the manuscript.

A thesis submitted for obtaining the candidate degree in technical sciences (Doctor of Philosophy) in specialty 05.13.23 - systems and means of artificial intelligence. – Lviv Polytechnic National University, Ministry of Education and Science of Ukraine, Lviv, 2020.

The dissertation research is devoted to increase the efficiency of forecasting of atmospheric air pollution parameters. As a task of environmental monitoring it has done on the basis of the developed and improved neurosimilar methods and instrumentalities of forecasting of atmospheric air pollution parameters.

In the *introduction* the relevance of the topic of the dissertation research has substantiated; the relationship with scientific programs, plans, topics has described; the purpose and tasks of the dissertation are formulated; methods of research have presented and scientific novelty has determined when solving the set tasks. Also, the practical significance of the obtained research results is reflected; lists of published papers on the topic of dissertation work and conferences, where the main results of the dissertation were tested, are presented.

In the *first section* the environmental monitoring systemwith its tasks are analyzed. The organization of observations, the evaluation of measured data and the prediction of the state of the environment are investigated. Therefore, in accordance with the tasks of environmental monitoring and taking into account the types and scales of the objects of observation, the analysis of a number of scientific Ukrainian and foreign works in a certain field are carried out and the study of local environmental monitoring of atmospheric air has grounded.

The major air pollutants has been investigated, among which agriculture, industrial enterprises, all modes of transport and other are studied. The low level of ealready equipped operating sources of dust and gas purification equipment leads to harmful emissions into the atmosphere of such pollution parameters as sulfur dioxide, nitrogen dioxide, carbon monoxide, methane, ammonia, dust, non-methane volatile

organic compounds, suspended solids and other. It is determined that in order to control the environment and to take timely environmental actions, it is necessary to predict the parameters of air pollution, including conditions of gaps in the data.

During analyzing the methods of filling the missing pollution parameters the investigation of the mean method and the regression modeling methods was justified. A number of regression modeling methods among which are the method of reference vectors, linear regression method with gradient descent, Adaptive Boosting, decision tree and neurosimilar structure of Sequential Geometric Transformation Model are researched.

The analysis of the forecasting of air pollution parameters models has showed that the models have to reflect the changes that occur in the air under the influence of human activity; to provide timely simulation subsystems with high quality environmental information; and to include a retrospective analysis of existing forecasts. In addition, exploration of heuristic, analytical and statistical approaches of the forecasting of atmospheric air pollution parameters was done. The neural network forecasting methods that can approximate nonlinear functions were analyzed: multilayer perceptron, neural network of radial basis functions, neural network of generalized regression, recurrent neural network, and recurrent neural network functions. The neurosimilar structure of the Sequential Geometric Transformation Model (SGTM) was researched, which was not previously used to solve forecasting problems in the context of partially missed atmospheric air pollution parameters.

Thus, it has been researched that the development of new methods of forecating of the atmospheric air pollution parameters is promising. And non-iterative methods based on the neurosimilar structure of the SGTM are investigated deeper in order to improve the accuracy of forecating of the atmospheric air pollution parameters on mobile and embedded devices.

Filling the missing atmospheric air pollution parameters using artificial neural networks or neurosimilar structures is due to doing their forecasting. So, in the *second section*, the development of a universal method of increasing the accuracy of finding the initial values by expanding the inputs with additional data samples has proposed. Two

methods of inputs expanding are described: one has based on the selection of compact points sets and another one has based on inverse-proportional quadratic functions.

It is researched that the developed method of filling the missing atmospheric air pollution parameters with inputs expansion by preliminary allocation of compact points sets of training sample increases the accuracy of filling of gaps in data fields. It is calculated that the average relative error of filling the mising atmospheric air pollution parameters based on neurosimilar structure of the SGTM is 21,9 %.

It is investigated that the use of the developed formation method of additional values of input data vectors by means of preliminary allocation of compact points sets provides increasing of accuracy of filling the missed air pollution parameters. It is proved that the average relative error of filling the missed concentrations of atmospheric air pollution parameters by nerosimilar structure of the SGTM is decreased by 8.5%. It is also established that the developed method does not work for all researched methods. For example, the application of the reference vector method with extended data sampling reduces the accuracy of filling in the missing attributes by 1.5%.

Also, in the second section, the procedure for improving the Yoh-Han Pao method of functional inputs expansion is described by using inversely proportional quadratic functions, which provides a reduction of errors by 4.3%.

In the *third section*, a method for correcting the error of short-term forecasting of air pollution parameters on the basis of a committee of neuro-like structures of different types is developed and investigated. The error correction is performed when prediction error of the nonlinear neuro-like structure is less than the prediction error of the linear neuro-like structure. The use of the error correction method reduced the forecasting error by 15% and increased the forecasting horizon by two days.

In addition, the method of constructing a of linear polynomials matrix coefficients that is created by their identification based on the results of learning the linear neuronal structure of the sequential geometric transformations model is further developed. It is proven that improved method reduces RAM memory costs until forecasting atmospheric air pollution parameters by 12,75 times.

In the *fourth section* neural network and neurosimilar models and methods for forecasting of air pollution parameters with gaps in the data (described in the previous sections) are developed. The developed software is implemented by using two languages - Java and Python. In the developed Python-framework parameter settings, training and application of neural network and neourosimilar models and methods of forecasting atmospheric air pollution parameters are performed. The Java-framework is used to implement methods that include the following modules: a set of the developed and improved methods; the interaction between two software projects; atmospheric air monitoring data processing module; module with classes, which contain methods for working with data tables; module for implementation of forecasting methods; module for calculating the accuracy of implemented neural and neurosimilar methods for forecasting air pollution parameters, including conditions of gaps in air monitoring data. The Java-framework also has a user interface for the ability to do forecast on mobile and embedded devices. Therefore, the Android Studio development environment was used to develop the software.

The set of developed methods to improve the accuracy of forecasting air pollution parameters with gaps in air monitoring data includes the following methods developed in Java-framefork: method of modeling additional values into input data vectors based on preliminary selection of compact points sets; improved Yoh-Han Pao expansion of inputs data vectors method based on rational fractions; a method of short-term forecasting of air pollution parameters based on error correction by the committee of neurosimilar structures of different types; and a method of constructing approximation linear polynomials by identifying their coefficients based on the results of training of the neurosimilar structure of the SGTM.

Consequently, based on the established forecasting errors of atmospheric air pollution parameters, the most effective methods are determined. It is experimentally proven that methods based on neurosimilar structures of the model of sequential geometric transformations show more accurate results of predicting of the atmospheric air pollution parameters and use less RAM than other searched models and methods.

Key words: environmental monitoring, pollution parameters, time series, neurosimilar structures of the sequentinal geometric transformations model, missing data, clustering, trend forecasting.

The list of PhD student's publications:

Articles in Ukrainian scientific professional editions:

1. Mishchuk O. Neural network with combined approximation of the surface of the response / O. Mishchuk, P. Vitynkyj // Science news of NTUU KPI: scientific and technical journal . — 2018. — № 2. — P. 18-24.
2. Mishchuk O. Methods of processing and filling of missing parameters in ecological monitoring data / O. Mishchuk, R. Tkachenko // Scientific Bulletin of UNFU. — 2019. — №29(6). — P. 119-122. doi: 10.15421/40290623
3. Mishchuk O. Multi-step forecasting of trends of atmospheric air pollution indicators // Scientific Bulletin of UNFU. — 2019. — №29(8). — P. 142-146.
4. Mishchuk O. Development of the method of forecasting the atmospheric air pollution parameters based on error correction by neural-like structures of the model of successive geometric transformations // Technology Audit and Production Reserves. — 2019. — № 6/2(50). — P. 22–26.

Article in the scientific periodical publication of another country:

5. Mishchuk O. The Accelerated Method of Filling Gaps in Data Using a Linear SGTM Neural-Like Structure / O. Mishchuk, R. Tkachenko, V. Pohrebennyk // International Journal of Science and Engineering Investigations (IJSEI). — 2019. — №8(91). — P. 154-159.

Proceedings of scientific conferences in serial foreign publications that included international scientometric databases:

6. Mishchuk O. Missing Data Imputation Through SGTM Neural-Like Structure for Environmental Monitoring Tasks / O. Mishchuk, R. Tkachenko, I. Izonin // Advances in Computer Science for Engineering and Education II. — International Conference on Computer Science, Engineering and Education Applications,

ICCSEEA 2019. — Springer. — Vol. 938. — P. 142-151. doi: 10.1007/978-3-030-16621-2_13 *(Scopus)*

7. Izonin I. SGD-Based Wiener Polynomial Approximation for Missing Data Recovery in Air Pollution Monitoring Dataset / I. Izonin, M. Greguš, R. Tkachenko, M. Logoyda, O. Mishchuk, Y. Kynash // Advances in Computational Intelligence. — 15th International Work-Conference on Artificial Neural Networks, IWANN 2019. — Vol. 11506. — P. 781-793. doi: 10.1007/978-3-030-20521-8_64 *(Scopus)*
8. Tkachenko R. A Non-Iterative Neural-Like Framework for Missing Data Imputation / R. Tkachenko, O. Mishchuk, I. Izonin, N. Kryvinska, R. Stoliarchuk // Procedia Computer Science. — The 14th International Conference on Future Networks and Communications, FNC 2019. — Elsevier. — Vol. 155. — P. 319-326. doi: 10.1016/j.procs.2019.08.046 *(Scopus)*
9. Izonin I. Recovery of Incomplete IoT Sensed Data using High-Performance Extended-Input Neural-Like Structure / I. Izonin, R. Tkachenko, N. Kryvinska, K. Zub, O. Mishchk, T. Lisovych // Procedia Computer Science. — International Workshop on Digitalization and Servitization within Factory-Free Economy, D&SwFFE-2019. — Elsevier. — Vol. 160. — P. 521-526. *(Scopus)*
10. Mishchuk O. One-step Prediction of Air Pollution Control Parameters using Neural-Like Structure Based on Geometric Data Transformations / O. Mishchuk, R. Tkachenko // Electronics and Information Technologies (ELIT-2019) : Proceedings of the XIth International Scientific and Practical Conference, Lviv, 16-18 September 2019. — Lviv. — P. 192-197. *(Scopus)*

Proceedings of international and national scientific conferences:

11. Mishchuk O. Neural-like structures of the geometric transformation model with combined surface approximation // Intellectual systems for decision making and problems of computational intelligence (ISDMCI-2018) : Proceedings of the XIVth International Scientific Conference, Zaliznyj Port, 21-27 May 2018. — Kherson. — P. 87-89.

12. Mishchuk O. Nonlinear inputs expansion of the neural-like structure of a sequential geometric transformation model // Computational methods and systems of information transformation (OMSIT-2018) : Proceedings of the Vth Scientific-Technical Conference, Lviv, 4-5 October 2018. — Lviv. — P. 126-129.
13. Mishchuk O. Restoring gaps in air pollution monitoring data using the neural-like structure of a sequential geometric transformation model // Computer modeling and optimization of complex systems (CMOCS-2018) : Proceedings of the IVth International Scientific-Technical Conference, Dnipro, 1-2 November 2018. — Dnipro. — P. 260-261.
14. Mishchuk O. Expansion of Neural-like structures inputs using combined approximation / O. Mishchuk, R. Tkachenko // Computer and Information Systems and Technologies (CSITIC-2019) : Proceedings of the IIIrd International Scientific Conference, Kharkiv, 23-24 April 2019. — Kharkiv. — P. 29-32.
15. Mishchuk O. Prediction of atmospheric air pollution parameters by linear neural-like structures // Building of information society: resources and technologies : Proceedings of the XVIIIth International Scientific and Practical Conference, Kyiv, 19-20 September 2019. — Kyiv. — P. 275-278.
16. Mishchuk O. Neural network method of forecasting the air pollution trend by carbon monoxide // Information Technologies and Automation (ITA-2019) : Proceedings of the XIIth International Scientific Conference, Odesa, 17-18 October, 2019. — Odesa. — P. 101-102.
17. Mishchuk O. Improving the accuracy of air pollution parameters forecasting // Computer modeling and optimization of complex systems (CMOCS-2019) : Proceedings of the Vth International Scientific-Technical Conference, Dnipro, 6-8 November 2019. — Dnipro. — P.129-130.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

Умовне позначення	Пояснення
АП	атмосферне повітря
БШП	багатошаровий перцептрон
ГДВ	гранично допустимий викид
ГДК	гранично допустима концентрація
ГК	головні компоненти
ЗР	забруднююча речовина
КМТ	компактна множина точок
МПГП	модель послідовних геометричних перетворень
НЕ	нейронний елемент
НС	нейронна структура
ПЗб	параметр забруднення
ШНМ	штучна нейронна мережа
AdaBoost	Adaptive Boosting (Адаптивний Бустинг)
API	рівень забруднення повітря
GIL	Global Interpreter Lock
GRNN	нейронна мережа узагальненої регресії
MAPE	середнє значення абсолютної похибки у відсотках
MAE	середнє значення абсолютної похибки
MAR	Missing At Random
MCAR	Missing Completely At Random

MNAR	Missing Not At Random
RPyC	віддалений Python-виклик
RBF	радіальна базисна функція
RMSE	середньоквадратична похибка
RMSE_M	середньоквадратична похибка у відсотках
SGDr	регресія на основі стохастичного градієнтного спуску
SVR	машина опорних векторів

ВСТУП

Актуальність роботи.

Велика кількість небезпечних викидів у повітряне середовище викликана функціонуванням промислових виробництв та автотранспорту, котрі містять низку постійно діючих джерел забруднення атмосферного повітря, що становить реальну загрозу для людини та всієї екосистеми планети. Тому у більшості країн світу здійснюють моніторинг навколишнього середовища із застосуванням рекомендацій Організації Об'єднаних Націй (резолюція №2286), враховуючи національні особливості.

Одним із завдань моніторингу довкілля є прогнозування розвитку певного екологічного процесу для забезпечення мінімального ступеня негативного впливу людської діяльності на повітряне середовище. Прогнозування параметрів забруднення атмосферного повітря, котре повсякчасно впливає на здоров'я людей, дозволяє випереджувати та вчасно реагувати на підвищення рівня викидів шкідливих домішок у повітря. До шкідливих домішок відносяться такі параметри забруднення повітря, як оксиди більшості важких металів, оксиди вуглецю, сірки, азоту, сажа, пил, вуглеводні, з'єднання свинцю та інші. Перелік параметрів забруднення атмосферного повітря визначається спеціальним державним органом та у кожній країні може бути різним. Завдання прогнозування параметрів забруднення довкілля полягає у передбаченні кожної окремо визначеної забруднюючої речовини, що викидається у повітряне середовище.

Особливості методів прогнозування параметрів забруднення повітряного середовища, в тому числі за допомогою засобів штучного інтелекту, зокрема штучних нейронних мереж, розглядаються у наукових роботах: К. А. Мальцева, В. Д. Погребенника, В. С. Джигирея, С. М. Дзюби, Л. Е. Чорнобая, О. В. Ничика, С. С. Харинцева, А. А. Севастьянова, М. Х. Салахова, Ф. Меканіка (F. Mekanik), М. Брауера (M. Brauer) та інших.

При виконанні завдання прогнозування параметрів забруднення атмосферного повітря виникає проблема неповноти даних, внаслідок пропусків окремих атрибутів у векторах виміряних параметрів забруднення повітряного

середовища під час виконання моніторингу довкілля. Це є перешкодою для прогнозування параметрів забруднення повітряного середовища та своєчасного реагування на збільшення рівня викидів шкідливих домішок у атмосферу.

Розв'язки задачі заповнення пропусків у даних, котрі передбачають інтелектуальний аналіз, підбір моделей та методів, та реалізацію цих методів сучасними інструментальними засобами подаються у роботах українських та зарубіжних авторів, серед яких: Р. М. Камінський, В. В. Пасічник, В. Б. Мокін, Н. Г. Загоруйко, О. О. Слабченко, Н. В. Кузнєцова, І. К. Зангієва, Б. П. Бочаров, Р. Дж. Літл (R. J. Little), П. Елісон (P. Elison), Дж. В. Грехем (J. V. Graham), Д. Ньюман (D. Newman), А. Карахаліос (A. Karahalios) та багато інших.

Високоточне прогнозування може бути реалізоване шляхом використання нейромережевих засобів та методів. Традиційні нейропарадигми, котрі базуються на основі методу рухомих часових вікон, не усувають негативний вплив шумоподібних коливань, що знижує точність прогнозів. Ситуативні зміни випадкових характеристик даних моніторингу довкілля вимагають періодичного перенавчання, що складно реалізувати для традиційних нейромережевих засобів з низькими характеристиками швидкодії навчання та складнощами налагодження параметрів. Нейроподібні структури на основі моделі послідовних геометричних перетворень (НС МПГП) мають певні біологічні аналогії лише в плані топології, але використовують математично обґрунтовані методи швидкого неітеративного навчання, що відкриває розширені можливості підвищення ефективності вирішення зазначеного завдання. Додаткові функції НС МПГП, зокрема можливість розкладу часових послідовностей на тренд і сукупність коливань, забезпечують додаткові можливості отримання важливої прогнозної інформації.

Концептуальна модель послідовних геометричних перетворень, що складає основу нейроподібної структури, має підвищену швидкодію та точність застосування. Однак вказана структура не використовувалася для розв'язку задач прогнозування в умовах невизначеності за рахунок наявних пропущених атрибутів у параметрах забруднення атмосферного повітря, для яких характерні такі особливості як швидка зміна в часі та наявність значних шумових складових.

Також не здійснювалось виділення лише тренду, де тренд в сумі складових є часовою послідовністю, що може бути ефективним саме в задачах прогнозування параметрів забруднення атмосферного повітря.

Разом з тим, розвиток моніторингу стану довкілля є важливим для контролю якості атмосферного повітря та своєчасного реагування на динаміку поведінки джерел викидів забруднюючих речовин з використанням різних пристроїв комп'ютерної техніки. Тому необхідною є розробка нових методів та засобів прогнозування параметрів забруднення підвищеної точності та швидкодії для користування на мобільних пристроях та контролерах.

Враховуючи описане вище, розробка перспективних методів та засобів прогнозування параметрів забруднення атмосферного повітря, в тому числі в умовах пропусків у даних, за допомогою нейронних структур моделі послідовних геометричних перетворень є актуальним завданням, що потребує розв'язку.

Зв'язок роботи з науковими програмами, планами, темами.

Дисертація виконана згідно з науковим напрямом кафедри інформаційних технологій видавничої справи – «Синтез та технології інтелектуального аналізу даних в гібридних інформаційних середовищах». Результати дисертаційної роботи використано під час виконання:

- держбюджетної науково-дослідної роботи за темою: «Нейромережева технологія захисту та передачі даних у реальному часі з використанням шумоподібних кодів» - номер державної реєстрації № 0119U002256;
- держбюджетної науково-дослідної роботи за темою: «Інформаційна технологія опрацювання персоналізованої медичної інформації» - номер державної реєстрації № 0119U002257.

Розроблені в результаті дисертаційного дослідження методи впроваджені в Здолбунівському відділенні АТ "РІВНЕГАЗ" для використання у робочому процесі підприємства під час виконання завдань моніторингу повітряного середовища.

Мета і завдання дослідження.

Метою дослідження є розроблення методів побудови нейроподібних структур з неітеративним навчанням для підвищення точності прогнозування параметрів забруднення атмосферного повітря в умовах пропусків у даних моніторингу довкілля.

Об'єктом дослідження є процеси прогнозування, зокрема в умовах пропусків у даних, під час виконання екологічного моніторингу для контролю забруднення повітряного середовища та своєчасного реагування на можливе виникнення надзвичайних ситуацій.

Предметом дослідження є нейроподібні структури моделі послідовних геометричних перетворень, їх методи навчання і застосування в режимі однокрокового та багатокрокового прогнозування, в тому числі в умовах пропущених даних моніторингу забруднення атмосферного повітря.

Завдання дослідження. Для досягнення мети дисертаційного дослідження було поставлено такі завдання:

1. Проаналізувати актуальні завдання моніторингу довкілля та особливості існуючих методів та засобів прогнозування параметрів забруднення атмосферного повітря, в тому числі в умовах пропусків у даних моніторингу повітряного середовища.
2. Розробити та апробувати метод короткотермінового прогнозування за допомогою використання нейроподібних структур моделі послідовних геометричних перетворень для збільшення горизонту прогнозування.
3. Розробити та дослідити метод формування додаткових вхідних атрибутів векторів даних для підвищення точності заповнення пропущених концентрацій параметрів забруднення атмосферного повітря.
4. Розвинути метод нейромережевої ідентифікації коефіцієнтів полінома, що відтворює функції навченої нейроподібної структури у режимі застосування для підвищення швидкості функціонування автономних мікроконтролерних пристроїв.

5. Удосконалити метод функціонального розширення вхідних векторів даних Йох-Хан Пао для зниження викидів в точках екстраполяції нелінійних поверхонь відгуку.
6. Розробити програмний засіб з набором бібліотек для виконання прогнозування параметрів забруднення повітряного середовища, зокрема в умовах пропусків у даних моніторингу атмосферного повітря.

Методи дослідження.

У роботі використано такі методи:

- метод середнього значення для заповнення пропусків у даних моніторингу повітряного середовища;
- методи машинного навчання на основі регресійного аналізу (Random Forest; Adaptive Boosting; SVR та SGDr);
- нейромережеві методи (багатошаровий перцептрон, нейронна мережа узагальненої регресії, радіально-базисна функція) вірогідного відновлення пропущених концентрацій оксидів азоту та карбону;
- метод підвищення вірогідності заповнення пропусків за допомогою попереднього виділення компактних множин точок та розподіл вхідних векторів даних до найближчого кластера, Евклідова відстань до якого є найменшою;
- метод нелінійного розширення входів нейроподібних структур моделі послідовних геометричних перетворень;
- методи наївного прогнозу та часових вікон для виконання однокрокового прогнозування параметрів забруднення повітря;
- метод короткотермінового прогнозування тренду забруднення повітряного середовища на основі корекції похибки за допомогою комітету нейроподібних структур різних типів;
- метод пришвидшеного прогнозування параметрів забруднення повітряного середовища на основі нейроподібної структури моделі послідовних геометричних перетворень та лінійних поліномів.

Наукова новизна одержаних результатів.

Під час розв’язання поставлених завдань отримано такі наукові результати:

вперше:

- розроблено метод уведення додаткових атрибутів – маркерів кластерів у вектори входів, що забезпечило підвищення точності заповнення пропущених показників параметрів забруднення атмосферного повітря;
- метод прогнозування параметрів забруднення атмосферного повітря за допомогою комітету лінійної та нелінійної нейроподібних структур для часткового коректування окремо додатних і від’ємних відхилень від точних значень, що забезпечило збільшення горизонту прогнозування;

удосконалено:

- метод функційного розширення входів Йох-Хан Пао шляхом застосування раціональних дробів, що забезпечило підвищення точності заповнення пропущених концентрацій параметрів забруднення атмосферного повітря за рахунок зниження викидів в екстраполятивних точках;

отримав подальший розвиток:

- метод побудови матриці коефіцієнтів лінійних поліномів, створеної шляхом їх ідентифікації за результатами навчання лінійної НС МПГП, що забезпечило підвищення швидкості прогнозування параметрів забруднення атмосферного повітря за рахунок зменшення затрат оперативної пам’яті мобільних пристроїв.

Практичне значення одержаних результатів.

Розроблений метод короткотермінового прогнозування часових послідовностей показників параметрів забруднення повітряного середовища за допомогою комітету лінійної та нелінійної нейроподібних структур забезпечив збільшення горизонту прогнозування тренду забруднення атмосферного повітря за рахунок часткового коректування окремо додатних і від’ємних відхилень від точних значень. Наприклад, для такого параметру забруднення повітряного середовища як чадний газ, похибка прогнозування зменшилась на 15 %, а горизонт прогнозування збільшився на два дні.

Розроблений метод заповнення пропущених атрибутів у параметрах забруднення повітряного середовища з розширенням входів за допомогою попередньої кластеризації вхідних векторів даних, відкинення аномалій та розширення входів тестової вибірки, забезпечив підвищення точності заповнення пропусків у даних моніторингу довкілля. Середня відносна похибка зменшилась на 8,5 % (для діоксиду азоту на 14,5 %, а для оксиду вуглецю на 2,5 %).

Дослідження показали, що розроблений метод підвищення точності заповнення пропусків у параметрах забруднення атмосферного повітря на основі розширення входів за допомогою попередньої кластеризації вхідних векторів даних спрацьовує не для всіх проаналізованих методів. Наприклад, розширивши вхідні ознаки векторів даних та застосувавши метод на основі опорних векторів, точність прогнозування погіршилась на 1,5 % (для діоксиду азоту на 1%, а для оксиду вуглецю на 2 %).

Удосконалений метод функційного розширення входів Йох-Хан Пао за допомогою застосування раціональних дробів забезпечив підвищення точності заповнення пропущених концентрацій параметрів забруднення атмосферного повітря на 2,6-6% в залежності від виду параметру.

Експериментально доведено, що метод пришвидшеного прогнозування параметрів забруднення атмосферного повітря, котрий базується на застосуванні нейроподібної структури моделі послідовних геометричних перетворень для визначення коефіцієнтів лінійних поліномів забезпечує зменшення часових затримок в режимі застосування. На прикладі чадного газу, встановлено, що час заповнення пропущених атрибутів цього параметру забруднення повітряного середовища становить 1,23 мілісекунд, що зменшило часові затримки прогнозування у 2-10 разів.

Отже, за допомогою розроблених методів та програмного засобу можна виконати одне із завдань моніторингу навколишнього середовища – прогнозування параметрів забруднення атмосферного повітря підвищеної швидкодії та точності на мобільних пристроях та мікроконтролерах для аналізу та прийняття природоохоронних управлінських рішень. Також розроблені методи та

програмний засіб прогнозування параметрів забруднення повітряного середовища забезпечують виконання заповнення пропусків у даних моніторингу довкілля підвищеної швидкодії та точності для ефективної оцінки прогнозованого стану навколишнього середовища.

Результати дослідження використані при виконанні держбюджетних науково-дослідних робіт «Нейромережева технологія захисту та передачі даних у реальному часі з використанням шумоподібних кодів» (№ 0119U002256) та «Інформаційна технологія опрацювання персоналізованої медичної інформації» (№ 0119U002257).

Особистий внесок здобувача.

Авторка самостійно отримала усі результати дисертаційної роботи. В публікаціях, написаних одноосібно здобувачці належать: розроблення методу короткотермінового прогнозування параметрів забруднення атмосферного повітря на основі нейронних структур різних типів [8, 15]; аналіз методів машинного навчання для прогнозування, в тому числі в умовах частково пропущених концентрацій параметрів забруднення атмосферного повітря [13]; дослідження методу введення додаткових ознак вхідних векторів, шляхом попереднього виділення компактних множин точок [11, 14]; удосконалення методу розширення входів Йох-Хан Пао за допомогою введення обернено-пропорційних квадратичних функцій [12]; розроблення методу підвищення точності прогнозування на основі методу корекції похибки за допомогою комітету нейронних структур різних типів [1, 16, 17]. В публікаціях, написаних у співавторстві здобувачці належать: розроблення методу розширення входів за допомогою виділення компактних множин точок [5]; розвиток та дослідження методу підвищення швидкості прогнозування параметрів забруднення повітряного середовища на основі лінійних поліномів [7]; реалізація та порівняння методів заповнення пропусків на основі нейроподібних структур [2, 3, 6, 9, 10]; виконання аналізу методу однокрокового прогнозування параметрів забруднення атмосферного повітря на основі нейронних структур моделі послідовних геометричних перетворень [4].

Апробація результатів дисертації.

Результати дисертаційної роботи були апробовані на наступних 12-ти наукових конференціях:

- XIV-й міжнародній науковій конференції «Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту» ISDMCI-2018 (м. Залізний Порт, 21-27 травня 2018 року);
- V-й науково-технічній конференції «Обчислювальні методи і системи перетворення інформації» ОМІСПІ-2018 (м. Львів, 4-5 жовтня 2018);
- IV-й міжнародній науково-технічній конференції «Комп'ютерне моделювання та оптимізація складних систем» КМОСС-2018 (м. Дніпро, 1-2 листопада 2018 року);
- IInd International Conference «Computer Science, Engineering and Education Applications» ICCSEEA-2019 (Kyiv, 26-27 January 2019);
- III-й міжнародній науково-технічній конференції «Комп’ютерні та інформаційні системи і технології» CSITIC-2019 (м. Харків, 23-24 квітня 2019);
- XVth International Work-Conference «Artificial Neural Networks» IWANN 2019 (Gran Canaria, 12-14 June 2019);
- XIVth International Conference «Future Networks and Communications» FNC-2019 (Halifax, 19-21 August 2019);
- XIth International Scientific and Practical Conference «Electronics and Information Technologies» ELIT-2019 (Lviv, 16-18 September 2019);
- XVIII-й міжнародній науково-практичній конференції «Побудова інформаційного суспільства: ресурси і технології» (м. Київ, 18-19 вересня 2019 року);
- XIIth International Scientific Conference «Information Technologies and Automation» ITA-2019 (Odesa, 17-18 October 2019);
- V-й міжнародній науково-технічній конференції «Комп'ютерне моделювання та оптимізація складних систем» КМОСС-2019 (м. Дніпро, 6-8 листопада 2019);

- International Workshop «Digitalization and Servitization within Factory-Free Economy» D&SwFFE-2019 (Coimbra, 4-7 November 2019).

Також результати дослідження були представлені на наукових семінарах кафедри інформаційних технологій видавничої справи Національного університету “Львівська політехніка” (2013-2014, 2017-2019).

Публікації.

За тематикою дослідження опубліковано сімнадцять наукових публікацій, в тому числі чотири статті у наукових фахових виданнях України [1, 7, 8, 9]; одна стаття у науковому періодичному виданні іншої держави [3], чотири матеріали конференцій у наукових серійних закордонних виданнях, що включено до міжнародних наукометричних баз [2, 4, 5, 6]; вісім тез та матеріалів доповідей на наукових конференціях [10-17].

РОЗДІЛ 1. ПРОГНОЗУВАННЯ ПАРАМЕТРІВ ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ ЯК ЗАВДАННЯ МОНІТОРИНГУ ДОВКІЛЛЯ

1.1. Аналіз системи моніторингу стану навколишнього середовища

Забруднення навколишнього середовища істотно впливає на зміни клімату; на стан вод, ґрунтів, озонового шару Землі та на всі живі організми та здоров'я людини зокрема. В навколишнє середовище потрапляє величезна кількість різноманітних токсичних речовин, яка щороку зростає. Забруднення довкілля відбувається під впливом природних та антропогенних факторів, спричинених діяльністю людини. Дослідження впливу цих факторів не можливе без спостереження за різноманітними параметрами біосфери, що є частиною моніторингу [1].

Згідно міжнародному стандарту ISO 4225, моніторинг − це багаторазове вимірювання змін певного параметра для спостереження протягом певного періоду часу та регулярне вимірювання рівнів забруднюючих речовин відносно визначеного стандарту з метою оцінки ефективності системи регулювання та контролю [2]. Окрім вимірювання рівнів шкідливих викидів та спостережень, моніторинг включає в себе оцінювання й прогнозування стану біосфери. В свою чергу, моніторинг навколишнього середовища − інтелектуальна система спостереження та контролю за довкіллям, що забезпечує збір інформації, її обробку, моделювання, оцінювання та прогнозування для прийняття управлінських рішень щодо раціонального використання природних ресурсів та щодо охорони довкілля вцілому [3].

Отже, метою системи моніторингу довкілля та контролю є не лише пасивна констатація фактів, а також їх аналіз, проведення експериментів, екологічне обґрунтування перспектив та вдосконалення системи моніторингу довкілля. А предметом моніторингу навколишнього середовища моделювання процесів та прогнозування стану біосфери, характеру впливу на неї природних та антропогенних факторів [4].

1.1.1. Завдання моніторингу навколишнього середовища

Моніторинг довкілля як напрям екології виник у другій половині XX ст., завданням якого було встановлення критеріїв оцінювання і виявлення меж стійкості екологічних систем. Його метою було отримання даних про стан та динаміку змін довкілля, створення баз даних і вибір об'єктів спостережень [5].

Моніторинг довкілля включає в себе завдання організації спостережень; наукового обґрунтування структури й методів спостережень за рівнем забруднення навколишнього середовища та станом біоти (сукупності живих організмів, що населяють досліджуваний район у визначений проміжок часу); вибір методик оцінювання та методів прогнозування стану довкілля; розробка обгрунтованих рекомендацій щодо управління станом довкілля [6, 7].

Основні ж завдання інтелектуальної системи моніторингу стану забруднення навколишнього середовища зображені на рисунку 1.1. [8].

Рис. 1.1. Система моніторингу стану навколишнього середовища

Як видно з рисунку 1, моніторинг навколишнього середовища включає в себе цикл таких завдань [8, 9]:

- спостереження за станом довкілля та факторами впливу на нього, що включає в себе вимірювання параметрів забруднення (ПЗб) біосфери;

- оцінювання фактичного стану навколишнього середовища, котре містить обробку виміряних параметрів моніторингу, отриманих на етапі спостереження;
- прогнозування стану навколишнього природного середовища і оцінювання прогнозованих ПЗб;
- останнім етапом циклу є виконання керуючих рішень по забезпеченню охорони навколишнього середовища від надзвичайних ситуацій (перевищення концентрації забруднюючих речовин (ЗР)).

Під час виконання завдань, моніторинг довкілля передбачає виконання наступних під завдань [5, 10]:

- дослідження обсягу впливу ЗР на довкілля та встановлення частини впливу людської діяльності на навколишнє природне середовище;
- визначення параметрів і джерел забруднення навколишнього середовища;
- виявлення критичних та надзвичайних ситуацій, що порушують екологічну безпеку.

Необхідність виконання всіх завдань та підзавдань зумовлює систему моніторингу довкілля, яка формується з блоків, що зображено на рис. 1. Блоки «Спостереження» і «Прогнозування» тісно пов’язані між собою, тому що прогнозування змін навколишнього середовища можливе лише за наявності повної інформації про його фактичний (виміряний) стан. Але не завжди є можливість отримати повні вибірки параметрів забруднення довкілля, тому виникають спотворення інформації через шумові перепони, поломки вимірювальних пристроїв чи навіть приховування [11, 12]. В результаті для виконання аналізу зібраних даних подаються дані з пропущеними параметрами забруднення довкілля, що в свою чергу спотворює прогнозовані дані моніторингу довкілля для прийняття природоохоронних рішень. Прогнозування потребує знання закономірностей змін стану довкілля та певну спрямованість прогнозу, яка зворотньо визначає структуру спостережень. Отримані під час спостережень чи прогнозування дані, оцінюють залежно від того, в якій сфері діяльності передбачається їх використання [13].

1.1.2. Рівні, види та підсистеми моніторингу довкілля

Налагодження системи моніторингу довкілля відповідно до його завдань сприяє виявленню екологічних небезпек, але ускладнює управління екосистемами, якщо порушене нормальне функціонування навколишнього середовища [14]. Тому згідно до завдань та масштабів об'єктів спостереження розрізняють рівні моніторингу довкілля, котрі досліджувалися українськими та закордонними вченими, спеціалістами у галузях ботаніки, геофізики, екології та описані у роботах Р. Є. Мунна (R. E. Munn), Ю. А. Ізраеля, І. П. Герасимова, Б. В. Виноградова, Н. Ф. Реймерса, М. А. Голубця [15 - 20].

Ю. А. Ізраель в наукових роботах доповідав про необхідність включення в програму моніторингу довкілля атмосферного повітря, атмосферних опадів та осадів, поверхневих вод, грунтів та біоти [16]. Найбільше значення дослідник приділяв чутливості представників кожного виду екосистеми до впливу низьких, близьких до фонових, концентрацій ПЗб, за рахунок чого стало можливим відображеня зміни коефіцієнту розмноження при впливі низьких рівнів ЗР.

В своїх роботах Герасимов І. П. заклав основи низки наукових напрямків, котрі проявились в рішенні задач, що відносяться до теми моніторингу навколишнього середовища. Географ за спецільність, вчений публікував наукові роботи про необхідність глибокого контролю за станом довкілля. Герасимов вперше поставив задачі для географів щодо рішення проблем моніторингу природного середовища та розглядав міжнародні програми по створенню глобальної системи моніторингу довкілля. Також, науковець детально розглянув та приділив особливу увагу ефективному контролю та достовірному прогнозуванню забруднення навколишнього середовища через безперервне розширення використання природних ресурсів [17].

Проаналізувавши згадані роботи, запропоновано наступний розподіл рівні моніторингу довкілля: глобальний моніторинг (охоплює всесвітню мережу наземних станцій, де вимірюються параметри забруднення); національний моніторинг (проводиться на території певної країни); регіональний моніторинг (проводиться в межах адміністративно-територіальних одиниць); локальний

моніторинг (проводиться на території окремих об'єктів: підприємств, площ, районів міста та ін) [19].

Досліджуючи дію природних і антропогенних факторів на неживі матерії та на живі організми, після аналізу роботи М. А. Голубця [20] пропонується розподіл моніторингу навколишнього середовища за такими ознаками: геологотехногенний моніторинг; соціальний моніторинг; техніко-економічний моніторинг; медико-біологічний моніторинг та екологічний моніторинг.

Концепція моніторингу навколишнього середовища полягає в необхідності здійснення повторюваних спостережень за елементами довкілля на протязі певного інтервалу часу за конкретними програмами [21]. На основі цієї концепції можна описати такі підсистеми моніторингу довкілля як [22]: моніторинг літосфери (передусім ґрунту); моніторинг приземного й верхнього шарів атмосфери; моніторинг атмосферних опадів; кліматичний моніторинг; моніторинг гідросфери (поверхневих вод суші, вод океанів, морів і підземних вод); моніторинг озонового шару; моніторинг океану.

У роботі реалізано методи прогнозування та заповнення пропущених параметрів у даних локального екологічного моніторингу атмсоферного повітря.

1.1.3. Спостереження за станом атмосферного повітря як підсистема моніторингу навколишнього середовища

Атмосфера є невід'ємною частиною нашої планети та є найважливішим природним ресурсом, який використовується для виробництва хімічних сполук, необхідних для життя людини [23]. Її речовинний склад є результатом діяльності екзогенних процесів, рослинності й океанів протягом мільярдів років. Хоча в процесі еволюції рослини, тварини та люди адаптувалися до хімічного складу та фізичних властивостей атмосфери, однак, техногенез, викликає зміни клімату та екологічних умов біосфери через постійні хімічні реакції [24]. В атмосферу попадають природні та антропогенні ЗР, що викликають зміну концентрації газів [25 - 27]. Наприклад концентрація таких як CO_2 та NO_2 постійно збільшується, що зображено на рисунку 1.2.

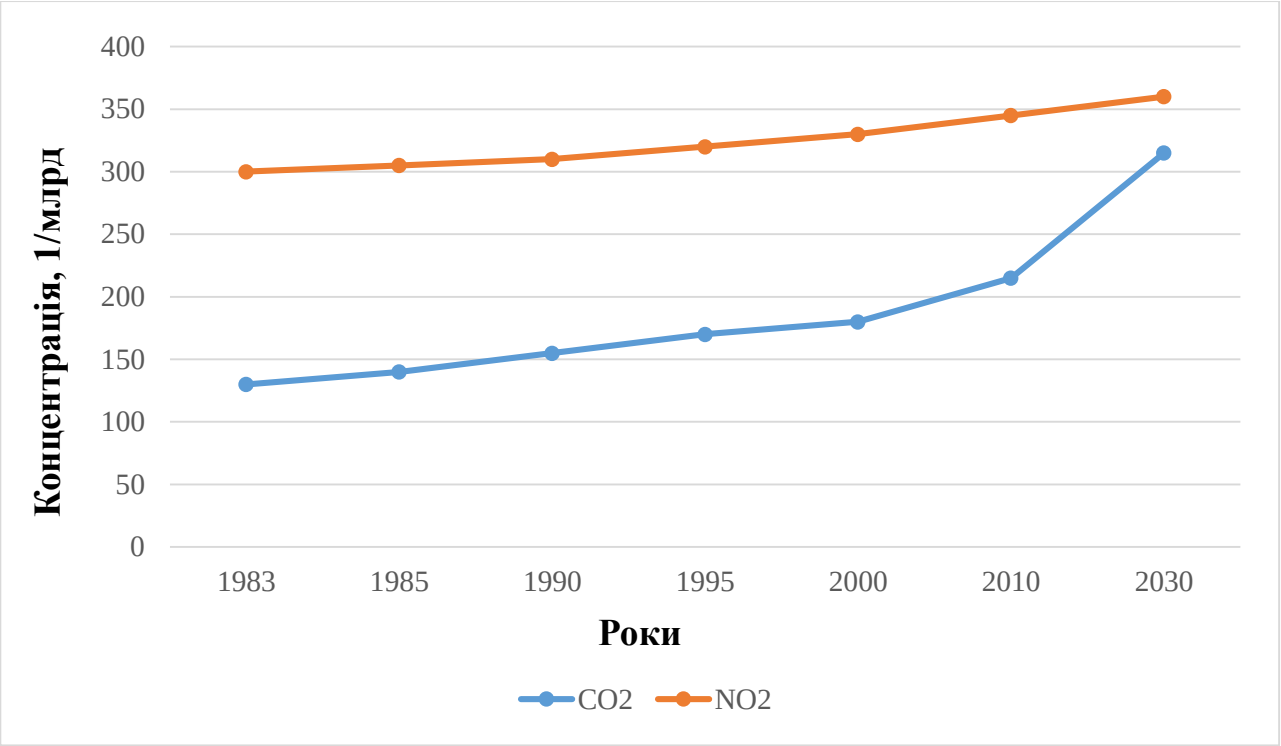


Рис. 1.2. Прогноз концентрації газів у Північній півкулі [24]

У тропічній зоні в атмосфері знаходиться 52% всього озону, що охоплює половину території Землі. Підтримка постійної концентрації озону є важливим завданням, оскільки концентрація озону в атмосфері зменшується через хімічні реакції, що проходять між викидами O_3 і NO [28].

Каталізаторами викидів NO_2 і NO в атмосферу є реактивні двигуни під час польотів літаків; продукти спалювання промислового палива; азотні добрива, що частково розпорошуються в атмосфері під час розпилення над грунтом. За рахунок розпорошування азотних добрив над грунтами, в атмосферу надходить близько 25-40% природного надходження оксиду азоту. Крім оксидів азоту в хімічних реакціях руйнування озону, беруть участь і оксиди водню, джерелом утворення яких також є промислові викиди. Також, руйнування озону відбувається в результаті хімічних реакцій озону з оксидом хлору, що є найбільш небезпечними для озону, джерелами яких є хлорвмісні сполуки – фреони ($CFCl_3$, CF_2Cl_2). Початок виробництва фреонів пов’язують з виготовленням холодильної техніки та у зв'язку з виробництвом різних аерозолів, – деодорантів, лаків, інсектицидів тощо [29]. В зв’язку з цим концентрація фреонів у атмосфері постійно зростає, що можна побачити на рис. 1.3.

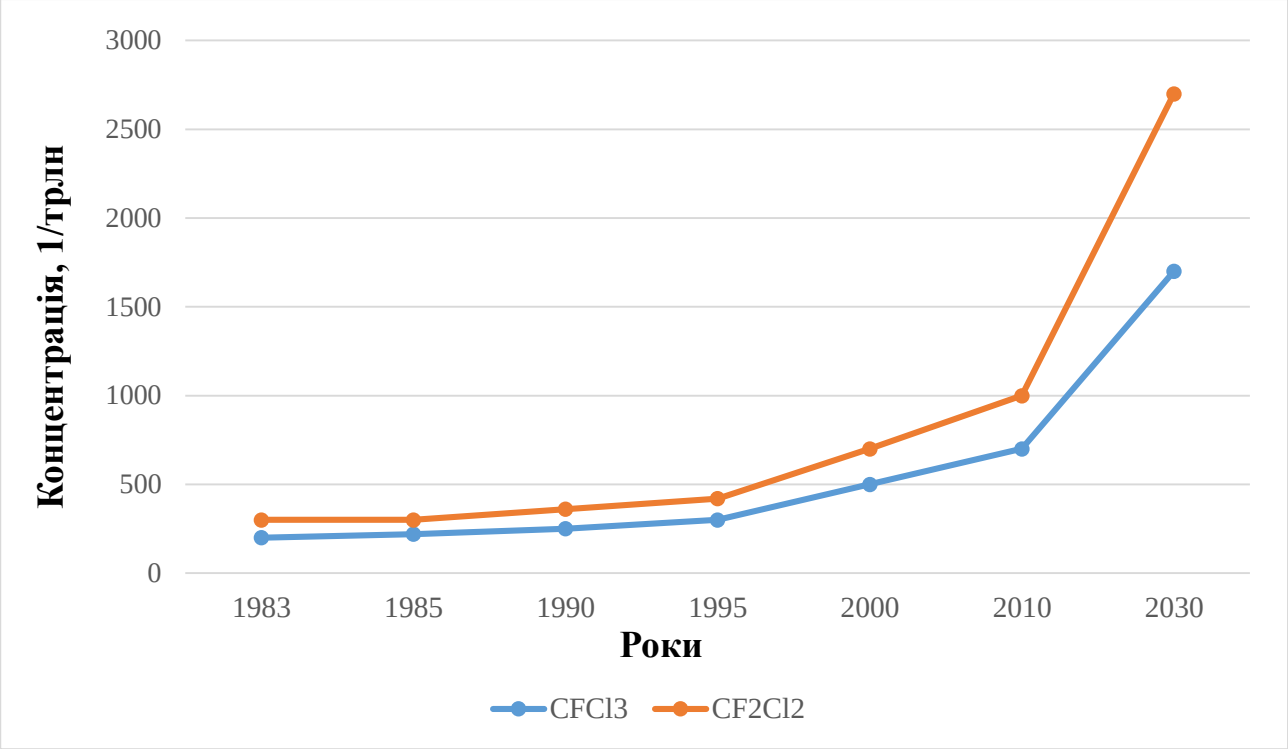


Рис. 1.3. Прогноз концентрації фреонів у Північній півкулі [24]

Основними антропогенними джерелами забруднення атмосфери належать: теплове та енергетичне устаткування; промислові підприємства, сільське господарство, всі види транспорту [30]. Особливо автотранспорт є найбільшим забруднювачем атмосферного повітря, тому що збільшення автотранспортних перевезень відслідковується кожного дня у світі. Це обумовлює зростання викидів відпрацьованих газів у забруднення повітряного середовища індустріально розвинених міст та областей, а тому збільшуються ризики для здоров'я населення, адже у цих газах налічується близько 100 різних компонентів, більшість з яких токсичні [31, 32].

Згідно регіональної доповіді Департаменту екології та природних ресурсів Київської обласної державної адміністрації за 2015 рік [33], протягом 2015 року в атмосферу надійшло 203,6 тисяч тон забруднюючих речовин від стаціонарних та пересувних джерел забруднення (без урахування викидів діоксиду вуглецю), що наведено у таблиці 1.1. [33], додаток Б. Це становить на 48,5 тисяч тон менше, ніж за 2014 рік. Основними напрямами зменшення надходження забруднюючих речовин в повітряне середовище є, насамперед виконання природоохоронних

заходів, котрі передбачаються обґрунтованими матеріалами про обсяги викидів забруднюючих речовин [34].

Крім того, основною причиною забруднення АП в Україні є низький рівень оснащення вже введених в експлуатацію та працюючих джерел викидів спеціальними газоаналізаторами та відсутність установок по вимірюванню основних газоподібних сполук: оксиду вуглецю, метану, діоксиду азоту, аміаку, пилу, неметанових летючих органічних сполук, зважених суспендованих частинок та інших. Викиди найпоширеніших параметрів забруднення атмосферного повітря наведено в таблиці 1.2 [33], додаток Б.

У регіональній доповіді Департаменту екології та природних ресурсів Київської обласної державної адміністрації за 2018 рік [35] зазначено, що основними забрудниками АП за той рік були підприємства постачання електроенергії, газу та кондиційованого повітря, адже їх викиди становили 68,5 % від загального валового обсягу викиду ЗР стаціонарними джерелами, що зображено на рисунку 1.4 [33, 35].

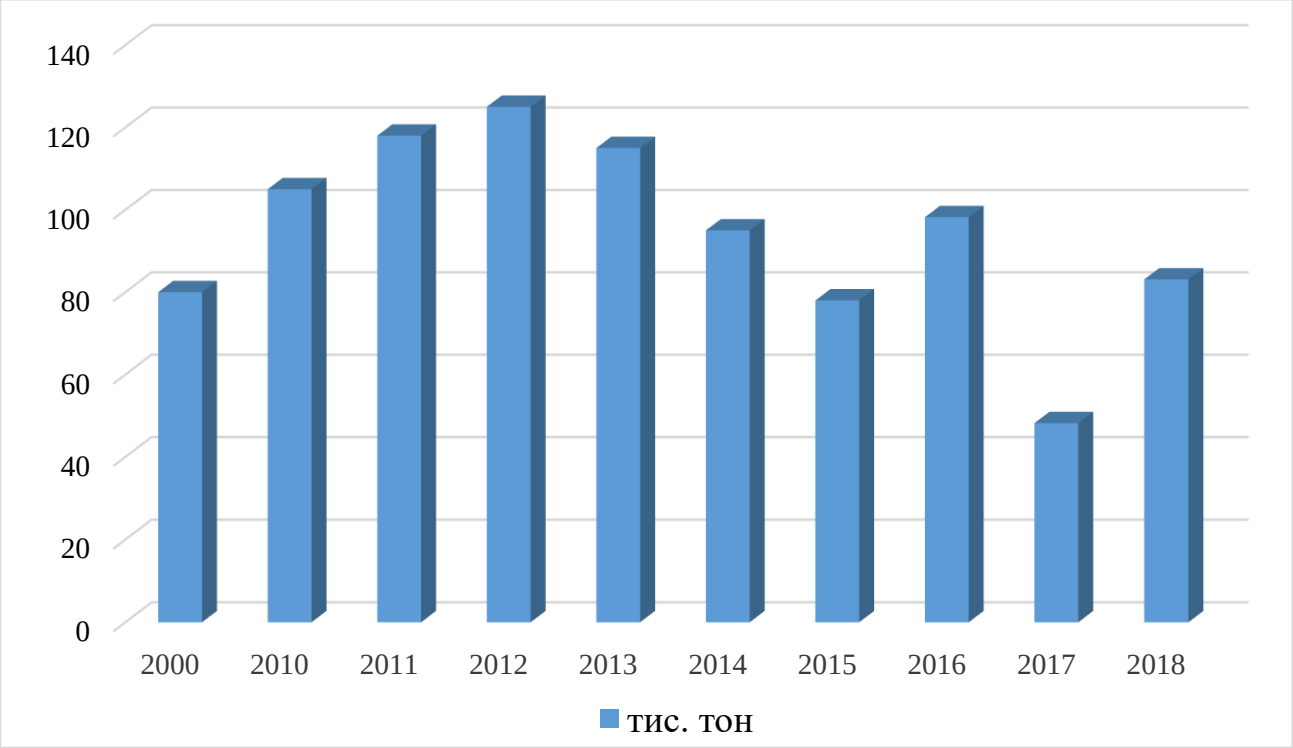


Рис. 1.4. Викиди ЗР в атмосферне повітря за 2000-2018 роки

1.2. Оцінювання фактичного стану атмосферного повітря за допомогою виміряних параметрів його забруднення

1.2.1. Нормування якості атмосферного повітря

Забруднюючі речовини потрапляють в організм людини через органи дихання, а частина шкідливих домішок, потрапляючи у легені, залишаються там [36]. ЗР, що поступають в атмосферу, утворюють стійкі завісі і містять оксиди більшості важких металів (вуглецю, сірки, азоту), сажу, пил, вуглеводні, з'єднання свинцю, та ін. Шкідливі домішки мають різну токсичну дію на організм людини і велике значення має тривалість дії забруднення [37].

У законодавстві України [38] вважається, що атмосферне повітря (АП) складається з суміші газів, що знаходиться за межами житлових, виробничих та інших приміщень. А для того, щоб оцінювати ступінь забруднення застосовують нормативи якості АП, що складаються з гранично допустимих максимальних величин вмісту ЗР у повітряному середовищі [39], серед яких визначають *гранично допустимий викид* (ГДВ) – технологічний норматив припустимого викиду забруднюючої речовини або суміші цих речовин в місці його виходу з устаткування та *гранично допустиму* концентрацію (ГДК) — норматив, що встановлює концентрації шкідливих речовин в одиниці об’єму повітря, котрі протягом певного часу майже не впливають на здоров’я людини.

Важливим є встановлення ГДК для забруднюючих речовин, для того, щоб використовувати ці нормативи при оцінці шкоди та обмеженні впливу на природні об'єкти. Регулювання викидів ЗР в повітряне середовище стаціонарними джерелами здійснюється для найбільш поширених і небезпечних ПЗб. Перелік параметрів забруднення атмосферного повітря встановлюється Кабінетом Міністрів України [38]. Для кожного окремого стаціонарного джерела забруднення атмосфери встановлюється ГДВ забруднюючої речовини або суміші ПЗб в атмосферне повітря. При цьому критеріями якості повітря на регіональному рівні, є максимально разові ГДК шкідливих речовин в повітрі населених пунктів, що вказані в таблиці 1.3. [40].

Оцінювання змін стану повітряного середовища дає змогу визначити можливі збитки та з’ясувати оптимальні умови людської діяльності [40]. Допустиме екологічне навантаження не спричиняє негативних наслідків для людей та не погіршує якості довкілля. Тому виконують оцінювання стану екосистеми за допомогою встановлення різниці між ГДК та фактично виміряними параметрами забруднення.

1.2.2. Обробка даних моніторингу повітряного середовища

Для виконання такого завдання моніторингу довкілля, як оцінювання фактичного стану АП, використовують різні методи отримання первинної та вторинної інформації. Отримання первинної інформації реалізуються безпосередньо через спостереження на стаціонарних чи пересувних джерелах забруднення. Отримання вторинної інформації полягає в аналізі даних, отриманих за допомогою первинної інформації [41].

Однією з найважливіших умов успішного оцінювання стану АП є наявність достовірної і повної інформації про виміряні протягом певного інтервалу часу параметрів забруднення повітряного середовища. Але достатньо часто дані моніторингу АП (як і довкілля) містять пропуски. Причинами появи пропущених параметрів у даних моніторингу АП може бути поломка приладів; шуми (несприятливі погодні умови); призупинення вимірювань під час вихідних днів, свят; помилки приладів вимірювання; пошкодження носіїв інформації; виконання недостатньої кількості вимірювань [42].

Оскільки мова йде про моніторинг забруднення довкілля, а саме повітря, без якого людина не може існувати, тому для якісного аналізу даних екологічного контролю важливу роль відіграє відновлення пропусків в таких даних. Проблеми обробки та аналізу пропущених параметрів в даних досліджені такими науковцями, як: Кузнєцова Н., Злоба Є., Яцків І., Зангієва І., Грехем Дж. (Graham J.), Карахайліос А. (Karahalios A.), Літтл Р. Дж. (Liitle R. J.), Ньюман Д. (Newman D.), Ван Бюрен С. (Van Buuren S.) та інших.

У своїх роботах Кузнєцова Н. В. [43, 44] розглядає методи обробки пропусків у даних враховуючи види, типи та формати даних, причини пропущених параметрів. Дослідниця описує спільні та відмінні риси існуючих методів обробки пропущених параметрів і визначає особливості їх застосування для заповнення пропусків у даних. Також вчена пропонує використання методів інтелектуального аналізу для заповнення пропущених параметрів у даних, зокрема методи регресійного аналізу. Злоба Є. та Яцків І. у [45] розглядають метод resampling як альтернативний до методу Бартлета, котрий вважають більш простим алгоритмічно при результуванні з такою ж якістю. Також дослідники описують метод заповнення пропущених даних за допомогою регресійного аналізу присутніх даних та визначають, що цей метод є більш перспективним способом обробки пропусків у даних порівняно з іншими методами.

Зангієва І. у роботі [46] запропонувала класифікацію методів обробки та заповнення пропущених параметрів у даних, котру взяли за основу інші дослідники. Вчена визначила, що методи заповнення пропусків поділяються на прості та складні, які в своєю чергу складаються з локальних та глобальних.

У роботах [47-48] Мокін В. Б. з іншими авторами розглядають питання створення пакету програм для автоматизації процесу вимірювання параметрів забруднення безпосередньо на місці контролю та наводять результати порівняльного аналізу засобів обчислювальної техніки.

Наприклад, в роботі Ньюмана [49] можна ознайомитися зі спробами узагальнення основ обробки пропущених даних у соціальних науках, де міститься п’ять простих для розуміння практичних вказівок з метою зменшити зміщення при наявності пропущених даних. Публікація передбачена для виконання кореляції, множинної регресії та моделювання структурних рівнянь з пропущеними параметрами у даних.

У роботах [50 - 52] Грехем описує та реалізує багаторазове заповнення пропусків. Але підбір алгоритмів обробки пропусків залежить від даних, від причин виникнення пропущених параметрів у даних, чи від області використання [53]. Тому виникає потреба у експериментальному дослідженні різних способів та

методів заповнення пропущених ПЗб атмосферного повітря для покращення результатів у даних екологічного моніторингу.

Проаналізувавши роботи Карахайліоса А., Грехема Дж. та Ван Бюрена С., сфомовано наступні способи оброблення пропущених параметрів забруднення у даних моніторингу атмосфреного повітря [50 - 54]: видалення векторів з пропущеними параметрами (є легким у виконанні, але неефективне, через можливість виникнення сильних зміщень); зважування повних спостережень для штучного досягнення запланованого обсягу вибірки (здійснюється із заданням початкових ваг, розділенням вибірки на підгрупи та обчисленням зважених рівних відгуків для кожної підгрупи); заповнення пропущених ПЗб.

Також, для того, щоб зрозуміти, як правильно обробити пропуски, необхідно визначити механізми їх формування. Розрізняють наступні три механізми формування пропусків [55]: Missing Completely At Random (MCAR — однакова ймовірність пропуску для кожного запису набору, тоді ігнорування/виключення векторів даних, що містять пропущені параметри не веде до спотворення результатів), Missing At Random (MAR — механізм формування не випадкових пропусків, а через деякі закономірності, коли ймовірність пропуску може бути визначена на основі іншої наявної в наборі даних інформації, що не містить пропусків [56]) та Missing Not At Random (MNAR — коли дані відсутні в залежності від невідомих факторів, а ймовірність пропуску описується на основі інших атрибутів, але інформація по них міститься у іншому наборі даних [57]).

У разі першого механізму пропусків (MCAR) застосування способу обробки пропусків, що полягає у виключенні з набору даних векторів з пропущеними параметрами, не призводить до істотного спотворення параметрів моделі. Проте видалення рядків призводить до того, що при подальших обчисленнях використовується не вся доступна інформація, стандартні відхилення зростають, отримані результати стають менш репрезентативними. У випадках коли пропусків в даних багато, це стає відчутною проблемою [58].

Крім того, в разі виникення пропуску внаслідок другого механізму (MAR), чи третього механізму пропусків (MNAR), зміщення статистичних властивостей

вибірки, значень параметрів побудованих моделей і збільшення стандартних відхилень збільшується [59]. Таким чином, незважаючи на широке поширення, застосування описаного способу обробки пропущених параметрів у даних для вирішення практичних завдань обмежена.

Інший спосіб обробки базується на ігноруванні пропусків в розрахунках. Такий спосіб часто застосовується за замовчуванням у різних пакетах обробки даних. Статистичні характеристики, такі як середні значення, стандартні відхилення, можна розрахувати, використовуючи всі непропущені значення для кожного з векторів [60]. За умови виконання гіпотези MCAR, застосування цього способу не призводить до суттєвого спотворення параметрів моделі. Перевага даного підходу полягає в тому, що при побудові моделі використовується вся доступна інформація. Головним же недоліком є виконання розрахунків без застосування всіх показників, що призводить до некоректних результатів [58].

Коли гіпотеза MCAR не виконується, використання двох згаданих способів обробки пропущених параметрів у даних призводять до суттєвих перекручень статистичних властивостей вибірки (середнього значення, медіани, варіації, кореляції та ін.). Також до їх недоліків відноситься і те, що, далеко не завжди виняток рядків в принципі прийнятний. Нерідко процедури подальшої обробки даних припускають, що всі рядки і колонки беруть участь у розрахунках (наприклад, коли пропусків у кожній колонці не дуже багато, але при цьому рядків, в яких немає жодного пропущеного поля мало) [61].

Тому, на практиці найбільше використовується третій спосіб обробки пропусків у даних, що передбачає заповнення пропущених параметрів. Недоліком такого способу є можливість істотного спотворення результатів, як і у інших способах [62]. До переваг даного способу відносяться: використання всього набору даних та явне використання інформації про пропущені значення.

Таким чином, проаналізувавши різні способи обробки пропусків у даних, у дисертаційній роботі для обробки пропущених параметрів забруднення атмосферного повітря виконується заповнення пропущених параметрів, котре включає в себе велику кількість досліджених методів.

1.2.3. Аналіз методів заповнення пропусків у даних моніторингу повітря

Вибір методу заповнення пропущених параметрів у даних моніторингу забруднення атмосферного повітря істотно залежить від методу аналізу даних, який буде використовуватися надалі, оскільки заповнення пропусків може змістити структуру вибірки. На рисунку 1.5. [45, 46] зображено класифікацію методів заповнення пропущених параметрів у даних на прості та складні.

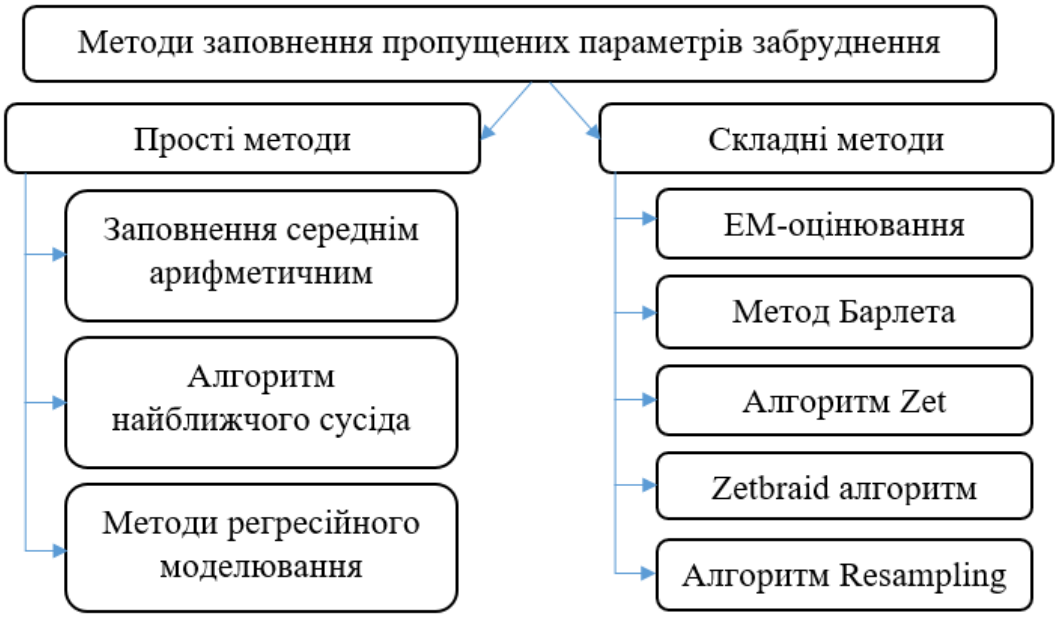


Рис. 1.5. Методи заповнення пропусків

На рисунку 1.5. можна побачити, що методи заповнення пропущених параметрів у даних моніторингу забруднення довкілля складаються з простих (неітеративних алгоритмів, заснованих на простих арифметичних операціях, відстанях між об'єктами, регресійному моделюванні) та складних (ітеративних алгоритмів з оптимізацією деякого функціоналу для відображення точнішого розрахунку значення, що підставляється на місце пропущеного параметру) [44, 63]. Серед простих алгоритмів заповнення пропущених параметрів розрізняють методи середнього значення (по n сусідніх точках, по певній даті), метод найближчого сусіда та методи регресійного моделювання [64-67].

Одним із проаналізованих складних методів заповнення пропущених параметрів у даних є *EM-оцінювання* (EM - expectation maximization), котрий

дозволяє не тільки відновлювати пропущені значення з використанням двох етапного ітеративного алгоритму, а й оцінювати середні значення, коваріаційні і кореляційні матриці для кількісних змінних [68]. Цей метод є ітераційною процедурою, що призначена для вирішення завдань оптимізації деякого функціоналу, через аналітичний пошук екстремуму функції [46].

Ще одним сучасним складним методом заповнення пропусків у даних є *метод Барлета* складається з таких етапів: підстановці замість пропусків початкових значень, проведенні коваріаційного аналізу цільової змінної і дихотомічного індикатора повноти спостереження за цільовою змінною [69, 70].

Наступний складний *метод ZET* полягає у підборі для кожного пропуску певного значення не з усієї сукупності повних спостережень, а з певної її частини, так званої компонентної матриці, яка складається з композитних векторів [71]. Компонентність деякого вектора або об'єкта є обернено-пропорційне значення Декартової відстані до цільового вектора (що містить пропущені показники ПЗб АП) в просторі, осі якого задані характеристиками об'єктів. За допомогою даних компонентної матриці можна побудувати функціональну залежність прогнозованого значення від відповідного значення в компетентній матриці, на основі якої, пізніше можна спрогнозувати значення пропуску [72].

Алгоритм ZetBraid є вдосконаленою версією алгоритму ZET та не містить цих недоліків через удосконалення процедури формування компетентної матриці методом плетіння котрий описаний у [73]. Зазначений алгоритм відноситься до локальних методів заповнення пропусків і для застосування алгоритму необхідно вирішити визначені завдання [74], одне з яких полягає у обчисленні відстані між рядками за формулою (1.1):

$$r_{ij} = P_{ij} + \sum_{k=1}^{n} b_k (a_{ik} - a_{jk})^2, \qquad (1.1)$$

де *Pij* - кількість стовпців, що мають пробіл в *i*-му або *j*-му рядку, b_k - ваговий коефіцієнт, значення якого залежить від того, чи входить *i*-й стовпець в компетентну матрицю.

Ще одним методом заповнення пропусків, котрий варто прийняти до уваги є *алгоритм resampling*, особливістю якого є те, що повторні вибірки витягуються з загальної сукупності, а псевдоповторні вибірки при виконанні ресемплінгу – з самої емпіричної вибірки [75]. Ресемплінг складається з чотирьох різних підходів до обробки даних: перехресна перевірка (cross-validation, CV), бутстреп (bootstrap), рандомізація (permutation) і метод "складного ножа" (jackknife). Під час використання підходу $r \times k$-кратної перехресної перевірки, вихідна вибірка випадковим чином розбивається r разів на k блоків рівної довжини [76]. При цьому генерується $r \times k$ значень відгуку $\tilde{y}$, і похибка перехресної перевірки на вихідній вибірці X^k розраховується за формулою (1.2):

$$S_{CV} = \sqrt{\frac{1}{r \times k} \sum_{i=1}^{r \times k} (y_i - \tilde{y}_i)^2} \qquad (1.2)$$

Виконати розбиття вихідної вибірки у на k блоків можна з використанням різних функцій мови R [77].

Складні методи заповнення пропусків у даних використання значень, які підбираються самим алгоритмом, а тому є затратними по часу та громісткими у використанні [78]. Прості ж методи заповнення пропущених елементів менш точні. Тому метою дисертаційної роботи є розробка такого методу заповнення пропущених параметрів у даних моніторингу забруднення, щоб був простим у використанні, при цьому швидший та точніший, ніж існуючі прості алгоритми. Для визначення ефективності розроблюваного методу заповнення пропусків у даних потрібно виконати експериментальне порівняння нового методу з наступними існуючими методами: методом середнього значення та регресійним моделюванням (метод опорних векторів, лінійна регресія з градієнтним спуском, AdaptiveBoosting, дерево рішень) [79-86].

Виконання методу середнього значення полягає у заміні середнім арифметичним значенням всіх пропущених параметрів. Зазначений метод часто є кращим апріорним припущенням для відсутніх значень, та має багато переваг, але не гнучкий до несезонних змін і може зробити лінії тренду менш помітною для

розпізнавання [79]. Для обчислення вибіркового середнього значення часто застосовують формулу (1.3), коли накопичують значення даних з послідовності:

$$\mu_k = \left(\sum_{i=0}^{k\Gamma 1} x_i\right) + x_k,\ k = 0, 1, \ldots, N-1,\ \tilde{\mu} = \frac{1}{N}\mu_{N\Gamma 1}, \qquad (1.3)$$

де N - кількість незалежних спостережень (розмір вибірки).

Якщо сума k величин виявляється істотно більше x_k, то помилка округлення може виявитися того ж порядку, що і значення x_k. У цьому випадку застосування обчислень з плаваючою крапкою може стати проблематичним [80].

Метод опорних векторів (Support Vector Machine – SVM) є методом максимізації математичної функції відносно наявного набору даних, котрий навчається на прикладах [81] та включає в себе: відділяючу гіперплощину, гіперплощину максимальної межі, м'яку межу та функцію ядра. Цей метод дозволяє обирати оптимальне розташування гіперплощини таким чином, щоб бути розташованою на максимальній відстані від елементів кожного з класів одночасно посередині визначеної зони, що відділяє між собою ці елементи [82]. Метод опорних векторів може також використовуватися як метод регресії (SVR), зберігаючи всі основні характеристики, а саме: мінімізування похибки та індивідуалізування гіперплану. Функція ядра обчислюється формулою (1.4):

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \qquad (1.4)$$

Метод лінійної регресії із стохастичним градієнтним спуском (SGDr – stochastic gradient descent regression) [83] є ітераційним методом, де на кожному кроці вектор ваг w змінюється в напрямку найбільшого убування цільової функції, тобто в напрямку антиградієнта за формулою (1.5):

$$\omega := \omega - \eta \nabla Q(\omega) \qquad (1.5)$$

де Q – швидкість навчання, що визначається експериментальним шляхом.

Реалізація алгоритму навчання методу лінійної регресії із стохастичним градієнтним спуском можливі два підходи: пакетний (на кожній ітерації алгоритму навчання розраховується нове значення вектора w) та стохастичний (на

кожній ітерації алгоритму випадковим чином вибирається тільки один об'єкт з навчальної вибірки) [83].

Метод Adaptive Boosting (AdaBoost) є мета-методом машинного навчання, в процесі навчання якого будується композиція з базових алгоритмів навчання для поліпшення їх ефективності, а також кожен наступний класифікатор будується по об'єктах, які погано класифікуються попередніми класифікаторами [84]. Після побудови визначеної кількості базових алгоритмів рекомендується виконати аналіз розподіл ваг об'єктів, а саме: об'єкти з найбільшою вагою, найвірогідніше, є шумовими викидами, які варто виключити з вибірки, після чого почати побудову композиції заново. Перевагами методу AdaBoost є простота реалізації та хороша узагальнююча здатність. Адаптивний бустинг використовуватися як універсальний метод фільтрації викидів перед застосуванням будь-якого іншого методу. Також алгоритм вимагає великих обсягів пам'яті для зберігання базових алгоритмів для бустингу та істотних витрат часу на обчислення задачі [84].

Метод на основі дерева рішень (дерева класифікацій чи регресійні дерева) – це спосіб подачі деяких правил в ієрархічній структурі, котра складається з елементів двох типів - вузлів (node) та листя (leaf). Вузли складаються з вирішальних правил, де виконується перевірка відповідності прикладів цього правила з деякого атрибуту навчальної множини, а листя є значеннями цільової змінної [85]. Основною сферою застосування дерев рішень є підтримка процесів прийняття рішень в статистиці, аналізі даних і машинному навчанні для створення моделі прогнозування значення цільової змінної на основі кількох змінних на вході. Дерева рішень використовуються для виконання таких завдань: класифікація (віднесення об'єктів до одного із заздалегідь відомих класів, де цільова змінна повинна мати дискретні значення); регресія (прогнозування числового значення незалежної змінної для визначеного вхідного вектора) та опис об'єктів (набір правил в дереві рішень, що дозволяє компактно описувати об'єкти замість використання складних структур) [86].

Описані методи заповнення пропусків досліджуються у дисертаційній роботі на даних моніторингу забруднення атмосферного повітря.

1.3. Завдання прогнозування параметрів забруднення атмосферного повітря

Серед методів аналізу, оцінювання та прогнозування стану довкілля розрізняють: емпіричне узагальнення (дослідження зв'язків між явищами та процесами об'єкту оцінки), метод аналогій (об'єкт дослідження підлягає оцінюванню відповідно до його типової моделі) та моделювання (побудова математичних, фізичних чи цифрових моделей) [87].

Під час виконання моніторингу нагромаджується база даних, котрі інформують про стан довкілля на певний час, основні процеси чи тенденції, що відбуваються в ньому. Ці відомості допомагають спрогнозувати його розвиток, передбачити надзвичайні ситуації природного та техногенного походження; спланувати науково обґрунтовані природоохоронні заходи для створення безпечних умов життедіяльності [88], а також відстежити антропогенні зміни у природі для досліджень природного середовища у динаміці, тобто оцінювання минулого, сучасного його станів, а також прогнозування змін його параметрів у майбутньому.

1.3.1. Моделі прогнозування параметрів забруднення повітря

Для здійснення прогнозування можливих змін довкілля в будь-якому масштабі (від глобального до локального) та будь якого виду (в тому числі і атмосферного повітря) потрібно володіти даними про сучасний стан навколишнього середовища та про плани господарської діяльності на визначеній території. Прогнозуваня складається з розроблення деякої моделі, що описує процес, результати якого передбачити потрібно передбачити [89, 90]. Моделі, спрямовані на виконання прогнозування локального екологічного забруднення атмосферного повітря, теоретично повинні відповідати тим самим вимогам, що і будь-яка математична модель. Важливим є здатність моделі узагальнювати доступні спостереження і на цій основі передбачати можливий розвиток подій з високим показником точності [91].

Будь-яка типова методика прогнозування включає такі необхідні елементи, як виконання передпрогнозованої орієнтації (визначення предмета, цілей, завдань і періоду попередження); створення передпрогнознозованого фону (збір та аналіз даних в інтервалі ретроспекції); формування вихідної базової моделі і конструювання пошукової моделі; підготовка, обґрунтування і прийняття необхідних рішень [92].

Аналізуючи методи прогнозування випливає їх класифікація на наступні групи [93 - 95]: евристичні (побудова інтуїтивних прогнозних моделей, які формуються експертами на основі цільової установки на виконання прогнозу, досвіду і знань експерта); аналітичні (коли відомі загальні закономірності розвитку процесу, його загальна структура, виражені функціональні зв'язки, та є контрольна вибірка, що дозволяє перевірити працездатність моделі); статистичні (основу яких складає формування стохастичних моделей прогнозування та передумовою застосування яких є наявність необхідних статистичних даних і відомостей, необхідних для визначення моделі прогнозу).

Прогнозування забруднення атмосферного повітря ділиться на довгострокове (від 1 до кількох місяців), короткострокове (від 4 діб до 1 місяця) та оперативне (від декількох годин до 3 діб) [96]. Для довгострокового прогнозування найчастіше застосовуються розрахункові (аналітичні, апроксимаційні) моделі. Для короткострокового та оперативного прогнозування зазвичай використовують статистичні моделі лінійної та нелінійної регресії [97], перевагою яких є простота реалізації і алгоритмізації. Але для оперативного прогнозування параметрів забруднення АП при аварійних викидах слід використовувати аналітичні моделі, що застосовуються для прогнозування поширення домішок від миттєвих точкових джерел [96].

В рамках короткострокового та оперативного прогнозування проводиться передбачення концентрацій найнебезпечніших забруднюючих речовин використовуючи модель множинної лінійної регресії за формулою (1.6) [98]:

$$q^P = b_0 + \frac{b_1}{v} + \frac{b_2}{t} \qquad (1.6)$$

де q^P - прогнозовані значення концентрацій, *b0, b1, b2*-коефіцієнти регресійної моделі, *v* - швидкість вітру, *t* - температура повітря (°С).

Прогнозування забруднення довкілля поділяється на прогнозування потенціалу забруднення та прогнозування концентрації [99]. Прогнозування концентрації параметрів забруднення складається з параметричних та непараметричних моделей. Параметрична модель полягає у визначенні параметрів рівнянь у відомій моделі для знаходження її виходу. Моделі, котрі включають в себе велику кількість історичних даних зазвичай є параметричними моделями. Однією з найпростіших параметричних моделей прогнозування є регресійна модель (модель тренду), де залежною змінною використовують досліджуваний атрибут, а незалежною – час чи номер вимірювання цього атрибуту [100].

1.3.2. Аналіз методів прогнозування параметрів забруднення повітряного середовища

Серед підходів до прогнозування параметрів забруднення АП розрізняють методи, побудовані на евристичних та статистичних алгоритмах, але вони є не достатньо точними. Тому варто розглянути методи прогнозування ПЗб атмосферного повітря з використанням часових рядів за допомогою штучних нейронних мереж (ШНМ), до яких належать [101 - 103]: одношарові перцептрони, мережі Adaline і Madaline, східчасті (ШНМ зустрічного поширення), багатошарові перцептрони (БШП), нейронні мережі радіальних базисних функцій (RBF), рекурентні нейронні мережі та інші, що здатні апроксимувати нелінійні функції.

Дослідженню ШНМ з апроксимацією поверхні відгуку присвячені роботи Харинцева С.С., Севастьянова А.А., Салахова М.Х. та інших в задачах прикладної спектроскопії [104]. У сфері екології, пошуки вирішення задачі прогнозування параметрів забруднення атмосферного повітря виконані у роботах Дзендзелюка О., Дзюби С., Прокопенко М., Гокхалє Ш. (Gokhale Sh.), Рахмана Н. (Rahman N.), Чорнобая-Луї Е. (Schornobay-Lui E.), Лі М. (Lee M.) та інших. У роботі [105]

Чорнобай-Луі Е. разом з іншими авторами описано розроблені моделі прогнозування параметрів забруднення атмосферного повітря за допомогою двох нейромережевих архітектур, а саме: багатошарової перцептронної (БШП) мережі та нелінійної авторегресивної екзогенної нейронної мережі для виконання управлінськиї дій щодо зниження концентрації вмісту визначених ПЗб атмосферного повітря.

Гокхалє Ш. У співавторстві з Харе М. у роботі [106] дослідили, що моделей якості повітря на основі детермінованих методів добре прогнозують концентрації забруднюючих речовин, що часто трапляються у вимірюваннях, але, як правило, вони не здатні передбачити "екстремальні" концентрації. На відміну від цього, моделі статистичного розподілу долають вищезазначене обмеження детермінованих моделей і прогнозують "екстремальні" концентрації. Однак екологічні збитки спричинені як "екстремальними" концентрацціями, так і середньою концентрацією забруднюючих речовин. Отже, автори зазначають, що модель прогнозування ПЗб АП повинна передбачати не тільки "крайні" діапазони, але й "середні" діапазони концентрацій забруднюючих речовин, тобто весь діапазон, а гібридне моделювання є однією з методик, яка прогнозує «весь діапазон» розподілу концентрацій забруднюючих речовин, поєднуючи детерміновані моделі, засновані на відповідних статистичних моделях розподілу. Тому у статті розроблено гібридну статистично-логістичну модель прогнозування розподілу концентрацій оксиду вуглецю (CO).

Рахман Н., Лі М., та Латфі М. у [107] представили модель часового ряду при прогнозуванні індексу забруднення повітря (API – Air Pollution Index) від трьох різних станцій: промислових, житлових та приміських районів. У цій роботі порівняуй сезонне авторегресивне інтегроване ковзне середнє, штучна нейронна мережа та три моделі нечітких часових рядів за допомогою середньої абсолютної та середньої квадратичної похибок. Застосування штучної нейронної мережі показало більш точні результати прогнозування API порівняно з авторегресивним інтегрованим ковзним середнім та нечіткими часовими рядами.

На основі аналізу [105 - 107] досліджень, виявлено, що точність моделей прогнозування за допомогою ШНМ є вищою, ніж під час використання статистичних моделей. Для дослідження методів прогнозуваня ПЗб АП було обрано такі ж методи регресійного аналізу як і для задачі відновлення пропусків у даних моніторингу забруднення АП, але ще додано наступні методи: БШП, Random Forest, радіально-базисну функцію (RBF), нейронну мережу узагальненої регресії (GRNN) та метод наївного прогнозу.

Одним із спрощених методів передбачення даних є *наївний прогноз*, котрий полягає в прогнозуванні низки значень за допомогою простої функції від значень прогнозованої змінної в близькому минулому. Найпростіша модель наївного прогнозу складається з припущення, що «завтра буде як сьогодні» [108].

Багатошаровий перцептрон (БПШ) – нейронна мережа, що складається з кількох шарів, які містять нейрони (точніше їх моделі) та дозволяють апроксимувати дуже складні нелінійні функції. Модель нейрона має кілька входів, кожен з яких має вагу, а нейрон, отримуючи сигнал, множить сигнали на ваги і підсумовує отримані величини, після чого передає результат до іншого нейрона або на вихід мережі [109].

Random Forest (Випадковий ліс) [110] є методом, який використовує ансамблевий метод навчання для класифікації та регресії. Дерева у випадкових лісах ведуться паралельно та між цими деревами не буває взаємодії. Метод Random Forest використовується для задач класифікації та прогнозування (регресії) окремих дерев. Випадковий ліс є найточнішим доступним алгоритмом навчання, дає оцінки, які змінні є важливими для класифікації та формує внутрішню неупереджену оцінку похибки узагальнення по мірі розвитку дерев рішень. Перевагами методу Random Forest також є ефективність застосування на великих базах даних, обробка тисячі вхідних змінних без видалення змінної та підтримка точності, коли значна частина даних відсутня [111]. Недоліком алгоритму є упередженість на користь тих атрибутів, що мають більше рівнів для даних, що включають категоричні змінні з різною кількістю рівнів.

Нейронна мережа радіально базисних функцій (RBF) є мережею, що використовує радіальні базисні функції у якості функції активації, виходом якої є лінійна комбінація радіальних базисних функцій входу та параметрів нейрона [112]. Мережі RBF застосовуються для апроксимації функцій, прогнозування часових рядів, задач класифікації та керування системами. RBF мережі зазвичай складаються з вхідного, прихованого, з нелінійною RBF функцією активації, та лінійного вихідного шарів. Вхід є вектором дійсних чисел, а вихід – скалярною функцією вхідного вектора, та розраховується за формулою (1.7):

$$\varphi(x) = \sum_{i=1}^{N} a_i \rho(\|x - c_i\|) \quad (1.7)$$

де N – кількість нейронів у прихованому шарі, c_i – центральний вектор для нейрона i, та a_i – це вага нейрона в лінійному виході нейронів [113].

Нейромережа узагальненої регресії (GRNN) - це варіація RBF, котра приназначена для решення задач узагальненої регресії, аналізу часових рядів, апроксимації функцій, прогнозування, класифікації. GRNN полягає в непараметричній регресії, коли кожен зразок навчання представляє собою середній для радіального базового нейрону. Характерною особливістю GRNN є висока швидкість навчання. Також до переваг цієї нейронної мережі належить можливість обробки шумів на входах при потребі невеликої кількості наборів даних та висока точність в оцінці, оскільки вона використовує функції Гаусса. Основним недоліком GRNN є те, що її розмір може бути величезним, що зробить його обчислювально дорогим [114].

Перераховані ШНМ володіють визначеними властивостями щодо часу навчання, складності налагодження, чи співвідношення характеристик точності та швидкодії. Також для кожної області використання потрібно виконувати додаткові експерименти для покращення результатів навчання та прогнозування, оскільки вибір алгоритмів апроксимації залежить від особливостей даних. Тому є потреба у дослідженні ШНМ для знаходження алгоритму, який найбільше задовольнятиме збільшенні швидкості та точності навчання та прогнозування.

1.4. Прогнозування параметрів забруднення атмосферного повітря за допомогою нейроподібної структури МПГП в умовах пропусків у даних

Галузь машинного навчання постійно розвивається та досліджуються нові алгоритми прогнозування на основі ШНМ та НС. Зокрема нейроподібні структури моделі послідовних геометричних перетворень (НС МПГП), котрі забезпечують виділення головних компонент даних, виділення тренду та коливань [115]. Для задач прогнозування параметрів забруднення атмосферного повітря важливим є дослідженння виділення лише тренду, де тренд в сумі складових є часовою послідовністю. Топологію НС МПГП подано на рис. 1.6. [116].

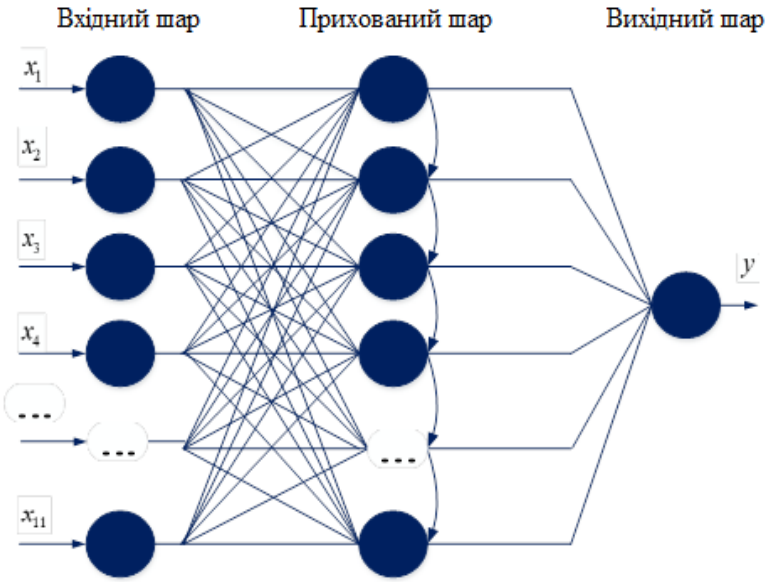


Рис. 1.6. Топологія НС МПГП

НС МПГП побудовані з використанням принципів просторового та часового розпаралелювання та можуть працювати в автоасоціативному режимі навчання (без вчителя) та в режимі з вчителем [117-120]. Під час використання НС МПГП виконується апроксимація та деякі геометричні перетворення над множиною об’єктів вибірки даних, які здійснюються в процесі навчання та застосування моделі. Базовими геометричними перетвореннями над об’єктами вибірки моделювання, що складаються з точок, є послідовна побудова нормальної гіперплощини та проектування точок на цю площину.

1.5. Формулювання актуальності, мети та завдань дослідження

Актуальність дисертаційного дослідження обгрунтовується важливістю прогнозування параметрів забруднення АП, що забезпечує можливість здійснювати моніторинг з випередженням та запобігати можливих негативних змін, завчасно приймаючи дії по зменшенню викидів шкідливих домішок у повітря. Оскільки практично не існує наборів даних без пропущених значень у них, актуальним такоє виконання заповнення пропусків у даних моніторингу повітряного середовища для підвищення точності прогнозуваня ПЗб.

Метою дослідження є розроблення швидкодіючих нейроподібних методів і засобів підвищеної точності для короткотермінового прогнозування параметрів забруднення атмосферного повітря в тому числі в умовах пропущених забруднюючих речовин у даних моніторингу повітряного середовища. Для досягнення поставленої мети, необхідно розв’язати такі завдання:

1. Розробити та апробувати метод короткотермінового прогнозування за допомогою використання нейроподібних структур моделі послідовних геометричних перетворень для збільшення горизонту прогнозування.

2. Розробити та дослідити метод формування додаткових вхідних атрибутів векторів даних для підвищення точності заповнення пропущених концентрацій параметрів забруднення атмосферного повітря.

3. Розвинути метод нейромережевої ідентифікації коефіцієнтів полінома, що відтворює функції навченої нейроподібної структури у режимі застосування для підвищення швидкості функціонування автономних мікроконтролерних пристроїв.

4. Удосконалити метод функціонального розширення вхідних векторів даних Йох-Хан Пао для зниження викидів в точках екстраполяції нелінійних поверхонь відгуку.

5. Розробити програмний засіб з набором бібліотек для виконання прогнозування параметрів забруднення повітряного середовища, зокрема в умовах пропусків у даних моніторингу атмосферного повітря.

ВИСНОВКИ ДО РОЗДІЛУ 1

1. Під час аналізу системи моніторингу навколишнього середовища та визначено, що її метою є аналізування, виконання експериментів, екологічне обґрунтування перспектив забруднення довкілля та контроль викидів ЗР. Встановлено, що предметом моніторингу довкілля є моделювання процесів та прогнозування забруднення біосфери, тенденції впливу на неї природних та антропогенних факторів. Досліджено, що завданнями моніторингу довкілля є організація спостережень, оцінювання та прогнозування стану довкілля, розробка обгрунтованих рекомендацій щодо природоохоронних дій. Згідно до завдань моніторингу навколишнього середовища та масштабів об'єктів спостереження проаналізовано низку наукових робіт, запропоновано розподіл моніторингу довкілля рівні, види і підсистеми, та обгрунтовано дослідження локального екологічного моніторингу атмсоферного повітря.

2. Аналіз антропогенних джерел забруднення атмосфери показав, що основними об'єктами викидів ЗР є теплове та енергетичне устаткування, промислові підприємства, сільське господарство, всі види транспорту, а особливо автотранспорт через постійне збільшення автотранспортних перевезень. Крім того, однією з основних причин забруднення АП є низький рівень оснащення вже працюючих джерел викидів пилогазоочисним обладнанням та відсутність установок по вловлюванню параметрів забруднення, серед яких шкідливими є викиди діоксиду сірки, діоксиду азоту, оксиду вуглецю, метану, аміаку, пилу, неметанових летючих органічних сполук, зважених суспендованих частинок та інших.

3. Досліджено, що вибір методу відновлення пропусків істотно залежить від методу аналізу даних, який буде використовуватися надалі, оскільки заповнення пропусків може змістити структуру вибірки. Серед вибраних методів заповнення пропущених параметрів забруднення атмосферного повітря виконано аналізування простих та складних методів та обгрунтовано реалізацію простих алгоритмів з метою менших затрат часу на навчання та виконання відновлення пропусків. Порівняно метод середнього значення та методи регресійного

моделювання, серед яких виділено метод опорних векторів, метод лінійної регресії з градієнтним спуском, Adaptive Boosting, дерево рішень та НС МПГП.

4. Проаналізувавши моделі прогнозування ПЗб атмосферного повітря встановлено, що розробка адекватних математичних моделей повинна відображати зміни, які відбуваються в природному середовищі під впливом людської діяльності; своєчасно забезпечувати підсистеми моделювання якісною інформацією про стан природного середовища і про параметри функціонування техносфери; включати в себе ретроспективний аналіз існуючих прогнозів з метою коригування математичних моделей, на основі яких вони були виконані.

5. Оскільки аналіз системи моніторингу довкілля показав, що одним із її функцій є прогнозування перспектив викидів ЗР у атмосферу, що ґрунтується на параметрах забруднення, виміряних на певний момент часу та в минулому, тому виконано розвідку евристичних, аналітичних та статистичних підходів до прогнозування забруднення атмосферного повітря. Серед існуючих методів прогнозування досліджено мережі Adaline та Madaline, штучні нейромережі зустрічного поширення багатошарові перцептрони, нейронні мережі радіальних базисних функцій, рекурентні нейронні мережі, та всі інші, котрі ґрунтуються на здатності апроксимувати нелінійні функції.

6. Також виконано аналіз нейромережевих методів прогнозування параметрів забруднення атмосферного повітря та методів на основі нейроподібних структур, серед яких розглянуто GRNN та НС МПГП. Досліджено, що НС МПГП, мають покращені характеристики, але не використовувалися для розв’язку задач прогнозування в умовах частково пропущених параметрів забруднення атмосферного повітря.

7. Таким чином, виконаний у першому розділі аналіз сучасних результатів досліджень показав, що для вдосконалення технологічних процесів моніторингу довкілля і їх автоматизації перспективним є розробка нових методів прогнозування параметрів забруднення атмосферного повітря на основі НС МПГП, в тому числі в умовах пропусків у даних моніторингу повітряного середовища з метою підвищення точності та швидкості застосування.

РОЗДІЛ 2. МЕТОДИ ВВЕДЕННЯ ДОДАТКОВИХ ВХІДНИХ ОЗНАК ВЕКТОРІВ ДАНИХ У ЗАВДАННЯХ ЗАПОВНЕННЯ ПРОПУСКІВ

Виконаний аналіз методів заповнення пропущених ПЗб у даних моніторингу АП в першому розділі показав важливість розробки нових методів відновлення пропусків за допомогою штучних нейронних мереж та нейроподібних структур, а саме за допомогою нейроподібних структур моделі послідовних геометричних претворень.

Проте не достатньо виконати відновлення пропусків у даних моніторингу забруднення АП за допомогою НС МПГП, а потрібно здійснити порівняння з іншими методами заповнення пропущених даних для підтвердження ефективності вибраного підходу. Тому, на рівні із застосуванням нейроподібної структури моделі послідовних геометричних претворень (НС МПГП), необхідним є виконання вибраних регресійних методів заповнення пропусків: дерева рішень, методу опорних векторів на основі регресії (SVR), лінійної регресії із застосуваням стохастичного градієнтного спуску (SGDr) та адаптивного бустингу (AdaBoost) [81 - 84].

Оскільки заповнення пропущених ПЗб у даних за допомогою штучних нейронних мереж чи нейроподібних структур відбувається за рахунок виконання їх передбачення, тому пропонується розроблення універсального методу підвищення точності знаходження вихідних значень за рахунок розширення вхідних ознак векторів даних.

Аналіз існуючих методів розширення входів обгрунтував доцільність дослідження двох видів розширення входів: лінійних, до яких належить апроксимаційні (згладжування даних), та нелінійних. Тому у роботі пропонується розроблення нового методу апроксимаційного розширення входів на основі виділення компактних множин точок, та удосконалення існуючого методу нелінійного розширення входів Йох-Хан Пао із виконанням введення обернено-пропорційних квадратичних функцій.

2.1. Метод введення додаткових вхідних ознак векторів даних шляхом попереднього виділення компактних множин точок

Для побудови ефективного методу заповнення пропущених значень слід проаналізувати особливості вірогідно відновлюваних концентрацій параметрів забруднення атмосферного повітря.

Як показали дослідження [121] (рис 2.1.) розподіл точок (векторів) реалізацій відповідає положенням гіпотези компактності, узагальненням якої є гіпотеза простої геометричної структури.

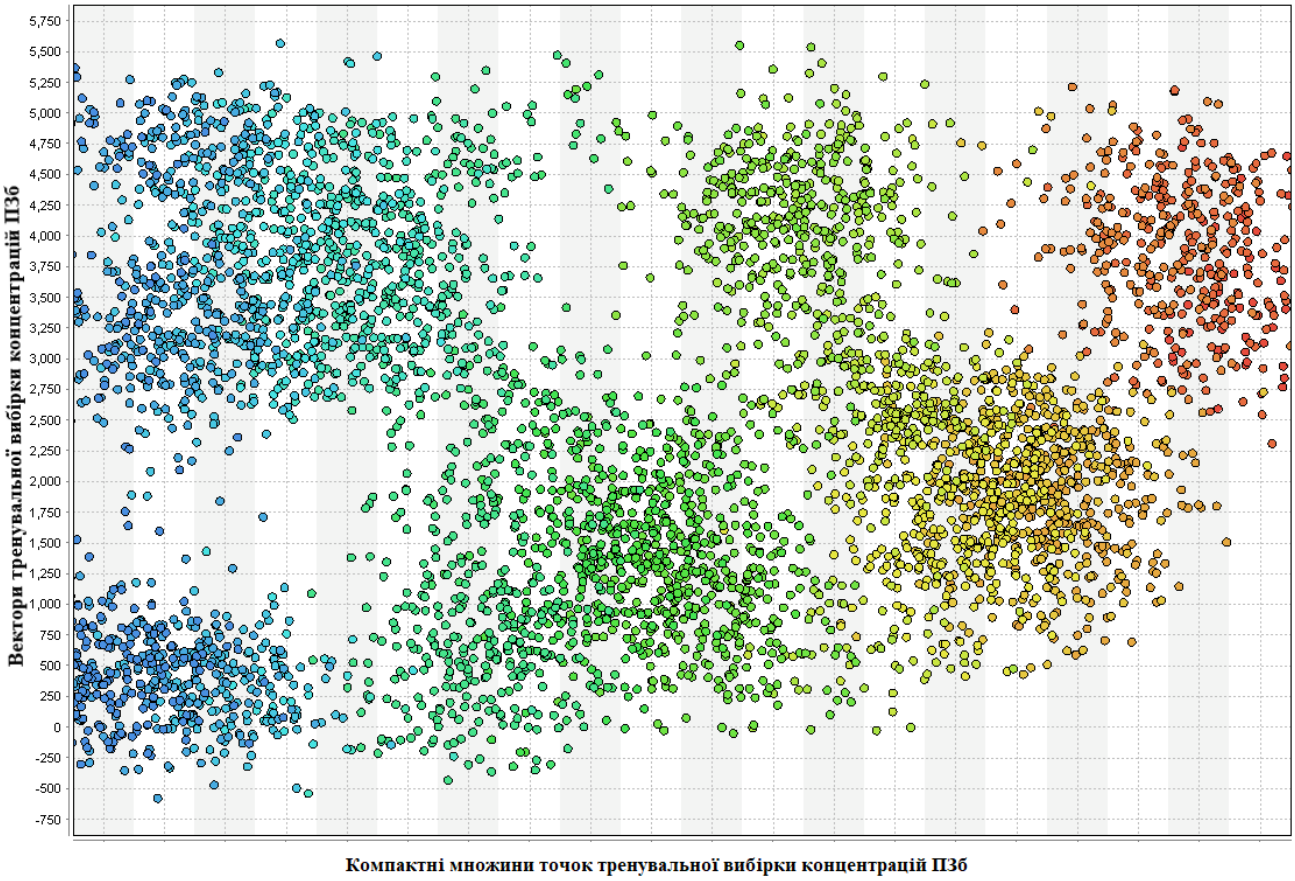


Рис. 2.1. Розподіл векторів тренувальної вибірки концентрацій параметрів забруднення атмосферного повітря по компактних множинах

На рисунку 2.1. проілюстровано виконання гіпотези компактності на досліджуваних даних моніторингу забруднення атмосферного повітря. Відповідно до неї, близьким по спільним характеристиках об'єктам в геометричному просторі ознак відповідають відокремлені безлічі точок, котрі мають властивості чіткого розподілу.

Отже, з рис. 2.1., видно, що дані моніторингу атмосферного повітря відповідають таким положенням гіпотези компактності: множини різних векторів концентрацій пересікаються в порівняно невеликому числі точок, а границі класів мають порівняно плавну форму та не мають глибоких виступів в межі інших класів. Врезультаті різні вектори можуть бути розділені досить простими гіперповерхнями за спільними в геометричному просторі ознаками: схильністю вздовж прямої, на колі, в сфері, по спіралі, на решітці і т.п.

Під час встановлення належності кожного вектора до відповідної компактної множини виникає можливість надання кожному з векторів тренувальної та тестової вибірок даних додаткового атрибуту – номера відповідного кластера. Звернемося до відомої теореми Ковера [122], яка стверджує що нелінійне перетворення завдання класифікації векторів в простір більш високої розмірності підвищує ймовірність лінійної роздільності образів.

Розглянемо сімейство поверхонь, кожна з яких ділить вхідний простір на K частин. Нехай множина X, що складається з N векторів $x_1, x_2, \ldots, x_N$, кожен з яких належить одному з K кластерів. Для кожного векторк $x \in X$ визначено функцію виду (2.1):

$$\phi(x) = [\phi_1(x), \phi_1(x), \ldots, \phi_{m_1}(x)]^T \qquad (2.1)$$

Якщо m_0 – розмірність векторів x, тоді функція $\phi(x)$ відображає точки m_0-мірного вхідного простору в простір розмірності m_1.

Розглянемо раціональні різноманіття r-го порядку у вхідному просторі розмірності m_0, тобто гіперповерхні описувані наступним рівнянням порядку r в координатах вхідного вектора x (2.2):

$$\sum_{0 \le i_1 \le i_2 \le \cdots \le i_r \le m_0}^{m_0} a_{i_1 i_2 \ldots i_r} x_{i_1} x_{i_2} \ldots x_{i_r} = 0, \qquad (2.2)$$

де вектор x_i - i-й компонент вхідного вектора x, що як зазвичай доповнений компонентом x_0=1 для надання рівнянню однорідної форми.

Для вхідного простору m_0 сума (2.2) складає $\frac{(m_0-r)!}{m_0!r!}$ доданків, що містять різні поєднання з r множників. Припустимо, що вектори $x_1, x_2, \ldots, x_N$, вибираються

незалежно, відповідно до ймовірнісного розподілу, що властивий вхідному простору, а всі можливі розбиття $X = \{x_i\}_{i=1}^{N}$ є рівноймовірними. Нехай $P(N, m_1)$ – ймовірність розбиття є ϕ-розділимою, якщо клас гіперповерхностей має m_1 степенів свободи, тоді (2.3):

$$P(N, m_1) = \left(\frac{1}{2}\right)^{(N-1)} \sum_{m=0}^{m_1-1} \binom{N-1}{m}, \qquad (2.3)$$

де $\binom{N-1}{m}$ – біноміальний коефіцієнт.

Рівняння (2.3) відображає суть теореми Ковера про розділення випадкових образів. Зрозуміло, що чим більша розмірність простору m_1, тим ближчою є ймовірність наближення $P(N, m_1)$ до 1 [122].

Отже, досягнення ефекту кращого розділення класів виникає за рахунок лінеаризації простору реалізацій за рахунок розширення його розмірності. Очевидно, що часткова лінеаризація простору підвищує точність розв'язку не лише завдань класифікації, але і регресії, до яких належить відновлення пропущених концентрацій параметрів забруднення АП.

Таким чином, в основі методу розширення входів є попереднє виділення кластерів та встановлення приналежності вхідних векторів даних до цих кластерів для лінеаризації простору. Метод введення додаткових ознак складається з сукупності процедур, на базі яких можна відновлювати пропущені концентрації ПЗб атмосферного повітря. Ця сукупність складається з послідовності наступних процедур:

Процедура 1. Розподіл вхідних векторів навчальної вибірки даних, які складаються з ПЗб атмосферного повітря, до визначених компактних множин точок. Для цього виконуються такі пункти:

- Дослідження методів каластерного аналізу та вибір методу виділення компактних множин векторів концентрацій ПЗб атмосферного повітря;
- виконання пошуку аномалій та їх відкинення (для зменшення впливу «екстремальних» викидів на результат заповнення пропущених атрибутів);
- вибір оптимальної кількості компактних можин точок (КМТ);
- розподіл векторів до деяких визначених оптимальних кластерів.

Процедура 2. Розширення вхідних векторів концентрацій навчальної вибірки даних моніторингу повітряного середовища.

Процедура 3. Розподілення вхідних векторів концентрацій ПЗб АП тестової вибірки даних до кластерів, визначених під час реалізації *Процедури 1*.

Процедура 4. Розширення вхідних векторів концентрацій ПЗб АП тестової вибірки даних моніторингу повітряного середовища.

Алгоритм розроблюваного методу розширення вхідних ознак векторів концентрацій ПЗб атмосферного повітря на основі попереднього виділення компактих множин точок зображено у вигляді блок-схеми на рисунку 2.2.

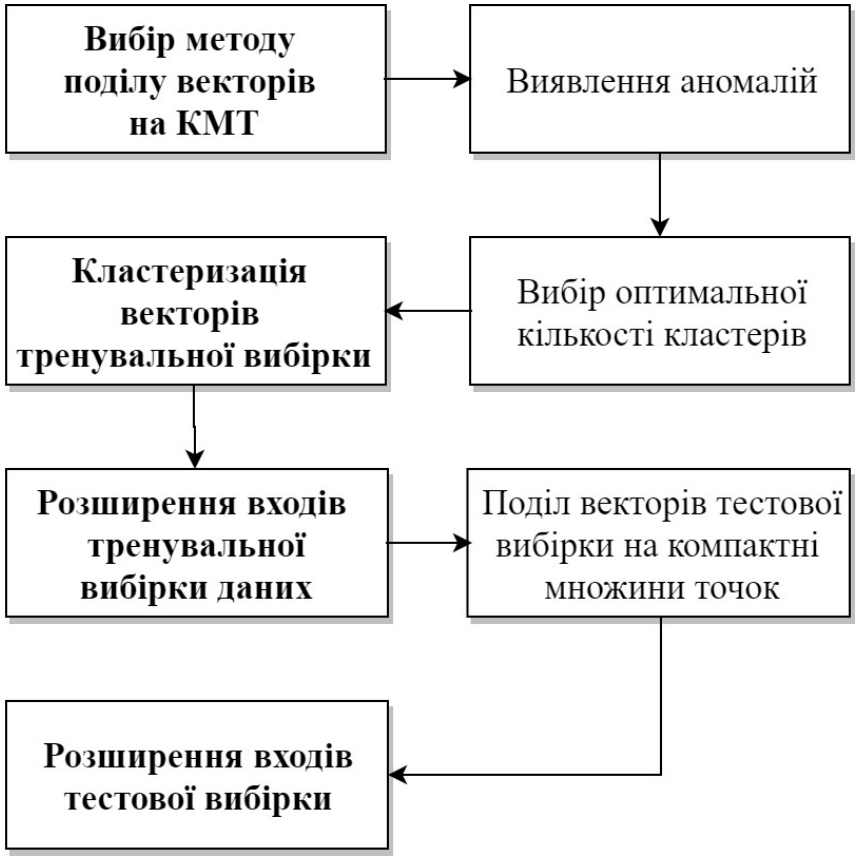


Рис. 2.2. Етапи реалізації методу введення додаткових атрибутів у вхідні вектори, шляхом встановлення їх належності до виділених кластерів

Кожен з етапів реалізації методу розширення входів на основі попереднього виділення кластерів та визначення приналежності векторів до кожного з них, скаладається з набору деяких процедур та кроків, котрі проаналізовано та описано в наступних параграфах.

2.1.1. Вибір методу кластеризації векторів даних моніторингу атмосферного повітря

Розширення входів векторів даних передбачає введення додаткових атрибутів, що складаються з 0 та 1 в тому атрибуті, номеру кластеру якого належить вектор, що розширюється. Для цього попереднім кроком потрібно виконати кластеризацію векторів.

Кластеризація передбачає розбиття множини об'єктів (точок чи векторів) на групи, які називаються компактними множинами точок (КМТ) [123]. Усередині кожного кластера повинні міститися «схожі» об'єкти, а об'єкти різних КМТ (кластерів) повинні бути найбільш відмінними. Оскільки перелік кластерів чітко не заданий, тому їх кількість визначається в процесі виконання деякого алгоритму. Застосування кластерного аналізу можна звести до таких етапів:

Eman 1. Відбір вибірки об'єктів для кластеризації.

Eman 2. Визначення змінних для оцінювання об'єктів у вибірці.

Eman 3. Обчислення значень міри «схожості» між об'єктами.

Eman 4. Застосування кластерного аналізу для створення КМТ.

Eman 5. Представлення результатів аналізу.

Обчислення значень міри «схожості» між векторами вхідних концентрацій параметрів забруднення АП на *Emani 3* виконуємо для кожної пари векторів за рахунок вимірювання «віддалі» в просторі між ними. Аналіз метричного простору, де віддаль $d\ (x_k^i, x_k^j)$ є ступенем близькості між елементами x і y, показав, що віддаль повинна відповідати таким обмеженням [124]:

1) $d(x_k^i, x_k^j) \geq 0$ – тобто, повинна бути невід'ємною;
2) $d(x_k^i, x_k^j) = 0$, якщо $x_k^i = x_k^j$;
3) $d(x_k^i, x_k^j) = d(x_k^j, x_k^i)$ – віддаль повинна бути симетричною.

Варто зазначити, що загальною метрикою визначення віддалей є метрика Мінковського [125], котра обраховується за формулою (2.4):

$$d_{ij} = r\sqrt{\sum_{k=1}^{n} |x_k^i - x_k^j|^r}, \qquad (2.4)$$

де r – параметр, що визначається дослідником для реалізації поступового зважування різниць деяких координат.

Якщо $r = 2$, то частковим випадком простору Мінковського є Евклідовий простір [126]. В такому випадку віддаль між точками обраховується як геометрична відстань в багатовимірному просторі (Евклідова віддаль) чи її квадрат (для додання більшої ваги віддаленішим один від одного об'єктів) за формулами (2.5) та (2.6):

$$d\left(x_k^i, x_k^j\right) = \sqrt{\sum_i^n (x_k^i - x_k^j)^2}, \tag{2.5}$$

$$d\left(x_k^i, x_k^j\right) = \sum_i^n (x_k^i - x_k^j)^2. \tag{2.6}$$

Якщо $r = 1$, віддаль між точками розраховується як відстань міських кварталів (Манхеттенська відстань) за формулою (2.7):

$$d\left(x_k^i, x_k^j\right) = \sum_i^n \left|x_k^i - x_k^j\right|. \tag{2.7}$$

У випадку коли r прямує до нескінченності, тоді віддаль між точками обчислюється за рахунок визначення відстані Чебишева [127], що обчислюється за формулою (2.8):

$$d\left(x_k^i, x_k^j\right) = max(\left|x_k^i - x_k^j\right|). \tag{2.8}$$

Оскільки більшість метрик розрахунку відстані між об’єктами зводяться деяким чином до Евклідової віддалі, тому для досліджень обрано визначення міри схожості векторів концентрацій параметрів забруднення атмосферного повітря за допомогою розрахунку Евклідової віддалі.

Для виконання *Emany 4*, вибираємо метод кластеризації, серед двох основних класифікацій: ієрархічних та неієрархічних методів, описаних у роботі [128]. Аналіз *ієрахнічних методів* показав побудову не одного розбиття вибірки на кластери, а системи вкладених розбиттів, тому на виході утворюється дерево кластерів, коренем якого є вся вибірка, а листям – найбільш дрібні кластери [129]. Варто зазначити, що результати таких методів зазвичай представляють у вигляді дерева – дендрограми чи графа.

Проаналізувавши *неієрархічні методи* кластеризації, виявлено, що на основі їх використання будується одне розбиття об'єктів на кластери. До неієрархчних методів відноситься метод квадратичної помилки, котрий для завдання кластеризації розглядається як побудова оптимального розбиття об'єктів на групи [130]. При цьому оптимізація виконується за рахунок зменшення середньоквадратичної помилки розбиття за формулою (2.9):

$$e^2(X, L) = \sum_{j=1}^{K} \sum_{i=1}^{n_j} \left\| x_i^{(j)} - c_j \right\|^2 \quad (2.9)$$

де c_j - «центр мас» кластера, тобто деяка точка (вектор) з середніми значеннями характеристик для певного кластера.

Найпоширенішим неієрархічним ітераційним методом кластерного аналізу є визначення квадратичної помилки за допомогою розрахунку k-середніх (k-means), що є простотим та наочним в реалізації.

Метод k-середніх (k-means) полягає в тому, що вектори даних довільно розбиваються на кластери, після чого ітеративно переобчислюються центри мас для кожного кластера, отриманого на попередньому кроці [131]. Наступним кроком вектори розбиваються на кластери знову відповідно до того, який з нових центрів виявився ближчим по обраної метриці. Метою цього методу є поділ n спостережень на k кластерів так, щоб кожне спостереження належало рівно одному кластеру, розташованому на найменшій відстані від спостереження.

Отже, на основі проведеного аналізу методів кластеризації, для виділення компактних множин точок вхідних векторів концентрацій ПЗб атмосферного повітря вибрано метод k-середніх (k-means). За допомогою цього методу вхідна множина даних розділяється на кластери так, щоб центри кластерів були максимально різні [132].

Реалізація методу k-середніх як етап розширення вхідних векторів конценртацій параметрів забруднення атмосферного повітря описано в параграфі 2.1.4.

2.1.2. Пошук аномальних концентрацій параметрів забруднення повітряного середовища

В даних часто трапляються аномальні концентрації параметрів, котрі значно відрізняються від інших значень. Аномальні значення можуть з'являтися з різних причин, серед яких: похибка вимірювання, спотворені дані та дійсні викиди (наприклад, дуже низька чи надзвичайно висока концентрація параметру забруднення).

Багато алгоритмів машинного навчання чутливі до діапазону та розподілу значень атрибутів у вхідних даних, тому наявність аномалій призводить до більш тривалого часу навчання, менш точних моделей і в кінцевому рахунку до гірших результатів.

Немає універсальних способів визначити аномальні викиди, тому важливо проінтерпретувати ряд даних так, щоб визначити, чи дійсно виміряне значення параметру забруднення атмосферного повітря є аномалією чи ні. Для цього потрібно розглянути та проаналізувати існуючі методи пошуку аномалій.

Аналіз методів пошуку аномалій показав наявність наступних методів [133]: аналіз екстремальних значень, метод проекції, імовірнісна та статистична моделі, лінійні моделі, моделі на основі близькості, інформаційно-теоретичні моделі, багатовимірне виявлення аномалії. Кожен із вказаних методів скаладається з деякого набору кроків для виконання аналізу даних та визначення екстремальних значень. Наприклад, метод аналізу екстремальних значень включає в себе такі кроки [134]:

Крок 1. Візуалізація даних, використовуючи гістограми чи графіки.

Крок 2. Знаходження гаусівського розподілу та пошук значення більшого за 2-3 стандартних відхилення від середнього чи в 1,5 рази більшого від першого чи третього квартилів.

Крок 3. Фільтрування ймовірних аномальних концентрацій з набору даних моніторингу забрудення атмосферного повітря та оцінка ефективності використаних моделей.

Проаналізовано, що порівняно простий у застосуванні та швидкий метод проекції має однаковий останній крок з методом аналізу екстремальних значень та складається з таких кроків [135]:

Крок 1. Використання методів прогнозування для узагальнення даних за двома вимірами (наприклад, PCA, SOM або відображення Самона).

Крок 2. Візуалізація даних та виявлення аномалій вручну.

Крок 3. Фільтрування ймовірних аномальних концентрацій з набору даних моніторингу забруднення атмосферного повітря та оцінка ефективності використаних моделей.

Суть методу середнього значення ряду та середньоквадратичного відхилення даного методу зводиться до того, що будь-які значення ряду, котрі відрізняються від середнього більше, ніж на два середньоквадратичні відхилення, є потенційними аномаліями. Поріг визначення аномалій задається формулою (2.10):

$$T = x_i \pm 2 \times \sigma \tag{2.10}$$

Варто зазчанити, що методи близькості [136] (екземпляри даних, які виділяються з множини значень, що визначаються кластерним аналізом чи визначенням найближчих сусідів) реалізуються за допомогою таких кроків:

Крок 1. Використання методу кластеризації для ідентифікації природних кластерів у даних (такого як алгоритм k-means).

Крок 2. Визначення центроїдів кластера.

Крок 3. Визначення відстані від даних до центроїдів кластера.

Крок 4. Фільтрування ймовірних аномальних концентрацій з набору даних моніторингу забруднення атмосферного повітря та оцінка ефективності використаних моделей.

Під час реалізацій методу розширення вхідних параметрів забруднення у даних моніторингу атмосферного повітря використано метод близькості на основі кластеризації тим же ж методом k-means, що вибрано для виділення компактних множин точок на попередньому етапі.

Отже, пошук аномалій виконується за допомогою наступних кроків:

Крок 1. Розрахунок для тренувальної та тестової вибіркок відстані від векторів до центрів кластера, в якому вони лежать, за формулою Евклідової відстані.

Крок 2. Візуалізуалізація даних та виявлення аномалій.

Крок 3. Аналіз візулізації.

Крок 4. Видалення аномальні векторів з тренувальної та тестової вибірок.

Під час пошуку аномалій методом близкості виконується розрахунок мінімальних Евклідових відстаней та візуалізація пропорцій кількості об'єктів щодо їх відстаней до центру кластерів (рис 2.3.).

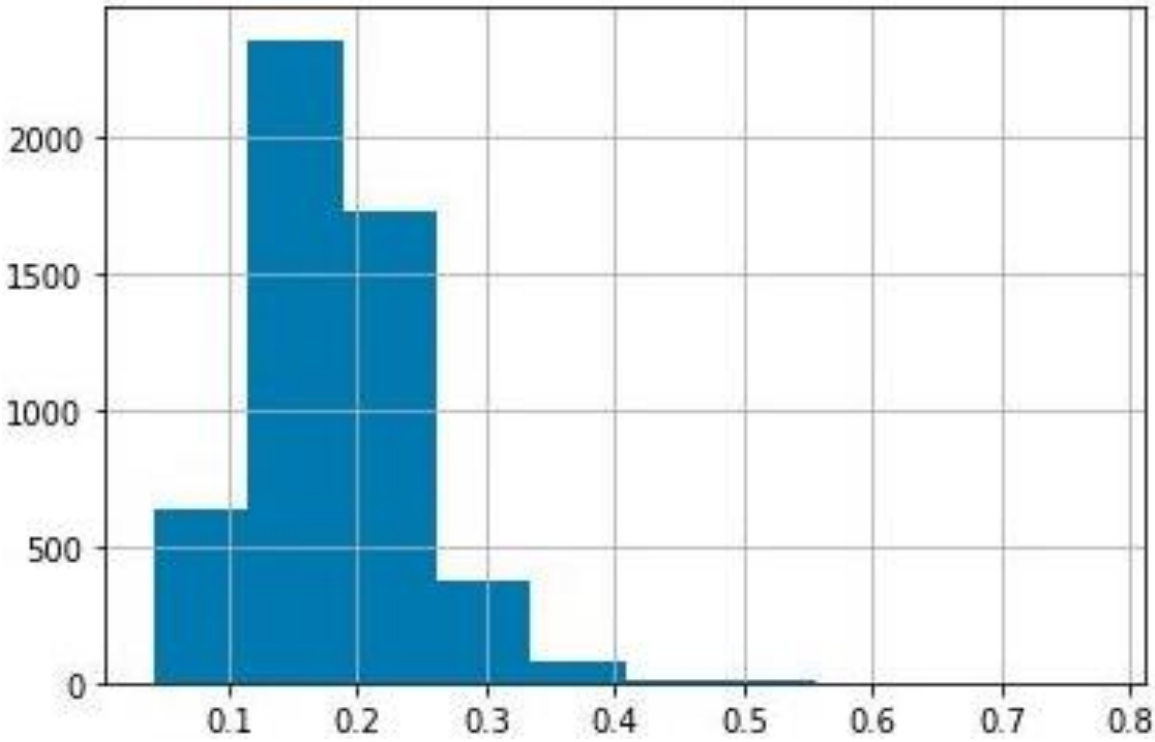


Рис. 2.3. Співвідношення кількості об’єктів (векторів концентрацій ПЗб АП) та відстаней до центрів кластерів

Аналізуючи візуалізацію на рис 2.3. можна зробити висновок, що вектори параметрів забруднення АП, в яких відстань до центру кластеру вища позначки 0.33 є аномаліями.

Наступним етапом розроблення методу розширення входів на основі попереднього виділення компактних множин точок є вибір оптимальної кількості кластерів.

2.1.3. Вибір оптимальної кількості кластерів

Виділення компактних множин точок (кластерів) включає в себе підбір оптимальної кількості кластерів шляхом їх об’єднання чи розподілу. Для визначення оптимальної кількості кластерів потрібно виконати аналіз існуючих метрик об’єднання між собою кластерів та обчислення «відстаней» між ними.

Аналіз метрики одиночного зв'язку [137] показав, що під час його використання, відстань між двома кластерами визначається віддаллю між двома найбільш близькими об'єктами (найближчими сусідами) в різних кластерах.

Оскільки метрика повного зв'язку полягає у визначенні найбільшої відстані між будь-якими двома об'єктами в різних кластерах (тобто найбільш віддаленими сусідами), тому ця метрика зазвичай працює дуже добре, коли об'єкти походять з окремих груп. Якщо ж кластери мають подовжену форму або їх природний тип є «ланцюжковим», то ця метрика є непридатною для виконання посталеного завдання [138].

Ще дві метрики: незважене попарне середнє та зважене попарне середнє [139], допомагають визначити відстань між двома різними кластерами як середню віддаль між усіма парами об'єктів в них. Проаналізувавши метрику незваженого попарного середннього, встановлено, що вона є ефективною, якщо об'єкти формуються у різних групах, проте визначено, що метрика працює так само добре і у випадках протяжних кластерів «ланцюжкового» типу. Метрика зваженого попарного середнього відрізняється від метрики незваженого попарного середнього тим, що при обчисленнях число об'єктів, котрі містяться у відповідних кластерах, використовується в якості вагового коефіцієнта. Тому дана метрика зваженого попарного середнього може бути використана, коли передбачаються нерівні розміри кластерів.

Виконавши аналіз метрик для обчислення відстаней між кластерами та для об’єднання їх між собою частіше за все користуються одиночним чи повним зв'язком. Тому в роботі обрано пошук оптимальної кількості кластерів метрикою повного зв’язку за допомогою розрахунку відстані найбільш віддалених сусідів.

Пошук оптимальної кількості кластерів складається з наступних кроків:

Крок 1. На вхід для аналізу подається деяка кількість кластерів;

Крок 2. Для кожної кількості кластерів відбувається розподіл векторів тренувальної вибірки даних моніторингу забруднення атмосферного повітря без врахування виходів;

Крок 3. Пошук суми відстаней до центру мас всіх векторів концентрацій параметрів забруднення атмоферного повітря у цьому кластері;

Крок 4. Візуалізація графіку співвідношення кількості кластерів та суми відстаней до центрів кластерів векторів (рис. 2.4.);

Крок 5. Вибір оптимального числа кластерів.

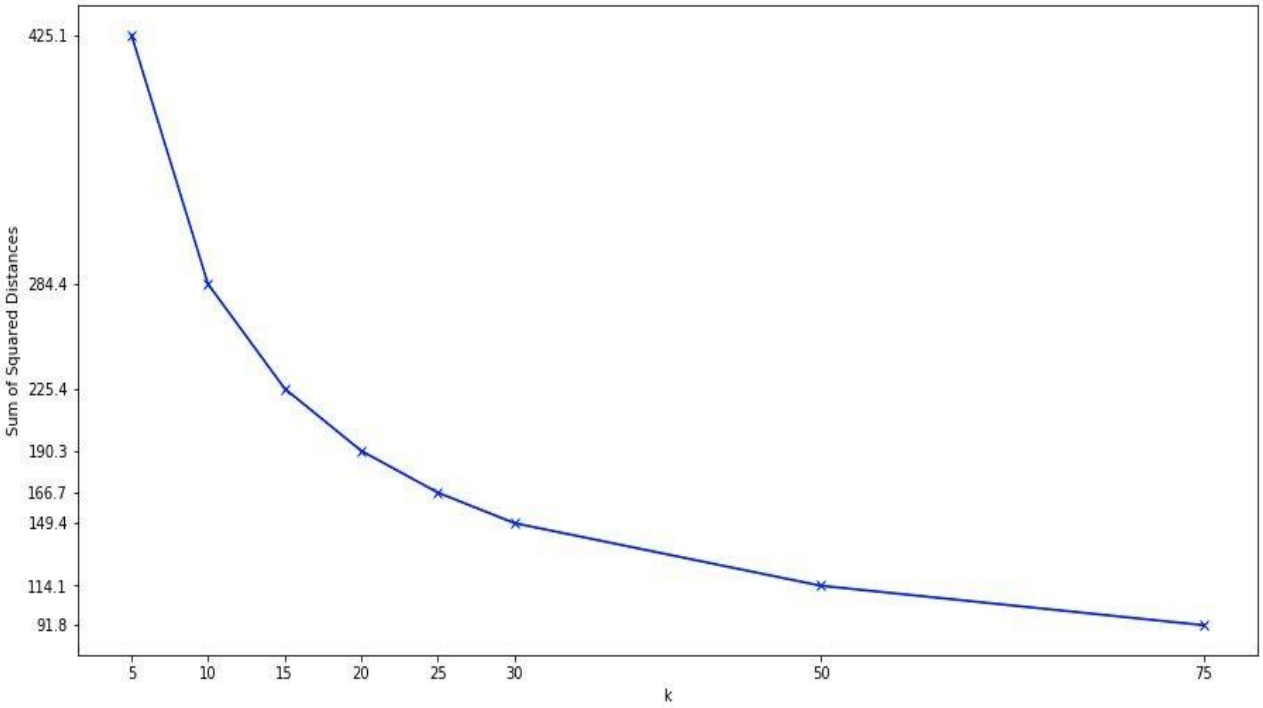


Рис. 2.4. Співвідношення кількості кластерів та суми квадратів відстаней до центрів мас кластерів векторів навчальної вибірки даних

З рис. 2.4. випливає, що найоптимальнішою кількістю кластерів є *k=20*, оскільки для наступного набору кластерів (*k=15*) відбувається різке підвищення суми квадратів відстаней.

Наступним етапом розширення вхідних параметрів забруднення повітряного середовища виконується виділення кластерів за допомогою методу k-means, подаючи на вхід знайдену оптимальну кількість кластерів.

2.1.4. Кластеризація векторів вибірок даних моніторингу атмосферного повітря

Для виконання поділу тренувальної вибірки параметрів забруднення атмосферного повітря $X = \{x_1, x_2, \dots, x_n\},\ x_i \in \mathbb{R}^d,\ i = 1, \dots, n$ на компактні множини точок (кластери) у попердніх параграфах обгрунотвано застосування методу k-середніх (k-means) та розраховано *k* оптимальних кластерів, $x_i \in N,\ k \leq n$.

Метод k-means розбиває тренувальну вибірку ПЗб атмосферног повітря на *k* наборів $S_1, S_2, \dots, S_k$, таким чином, щоб мінімізувати суму квадратів відстаней від кожної точки кластера до його центру (центр мас кластера). Введемо позначення $S = \{S_1, S_2, \dots, S_k\}$, тоді дія методу k-means є рівносильною пошуку:

$$arg\ min \sum_{i=1}^{k} \sum_{x \in S_i} \rho(x, \mathrm{c}_i)^2, \tag{2.11}$$

де c_i - центри кластерів, $i = 1, .., k$, $\rho(x, \mathrm{c}_i)$ - функція відстані між x і c_i.

Метод k-means складається з наступної послідовності кроків [140]:

Крок 1. Ініціалізація кластерів, котра виконуєтсья за рахунок вибору довільної множина точок $\mathrm{c}_i,\ i = 1, \dots, k$, що розглядаються на цьому кроці як початкові центри кластерів: $\mathrm{c}_i^{(0)} = \mathrm{c}_i,\ i = 1, \dots, k$.

Крок 2. Розподіл векторів концентрацій параметрів забруднення атмосферного повітря по найближчих визначених оптимальних кластерах: $\forall x_i \in X,\ \ i = 1, \dots, n: x_i \in S_j \Leftrightarrow j = arg\ min_k\ \rho(x_i, \mathrm{c}_k^{(t-1)})^2$. Для цього кроку між векторами $x_i \in X,\ i = 1, \dots, n$ і центрами кластерів $\mathrm{C}_1, \dots, \mathrm{C}_k$ потрібно обчислити Евклідові відстані за формулою 2.6 описаною у параграфі 2.1.1. Таким чином, цей крок сладається з k_n обчислень відстаней між d-мірними векторами [141].

Кожен з векторів концентрацій ПЗб атмосферного повітря відноситься до того кластеру, відстань до центру якого для нього мінімальна:

$$S_i^{(t)} = \{x_p: \left\| x_p - \mu_i^{(t)} \right\|^2 \leq \left\| x_p - \mu_j^{(t)} \right\|^2\ \ \forall j = 1, \dots, k\}, \tag{2.12}$$

де кожен вектор x_p співвідноситься єдиному кластеру $S^{(t)}$.

На рисунку 2.5. зображені вхідні вектори $x_1, \dots, x_n$ та центри мас кластерів $C_1, \dots, C_k$, обчислені раніше (на кроці ініціалізації, якщо йдеться про першу ітерацію алгоритму, або на кроці перерахунку центрів кластерів попередньої ітерації в іншому випадку).

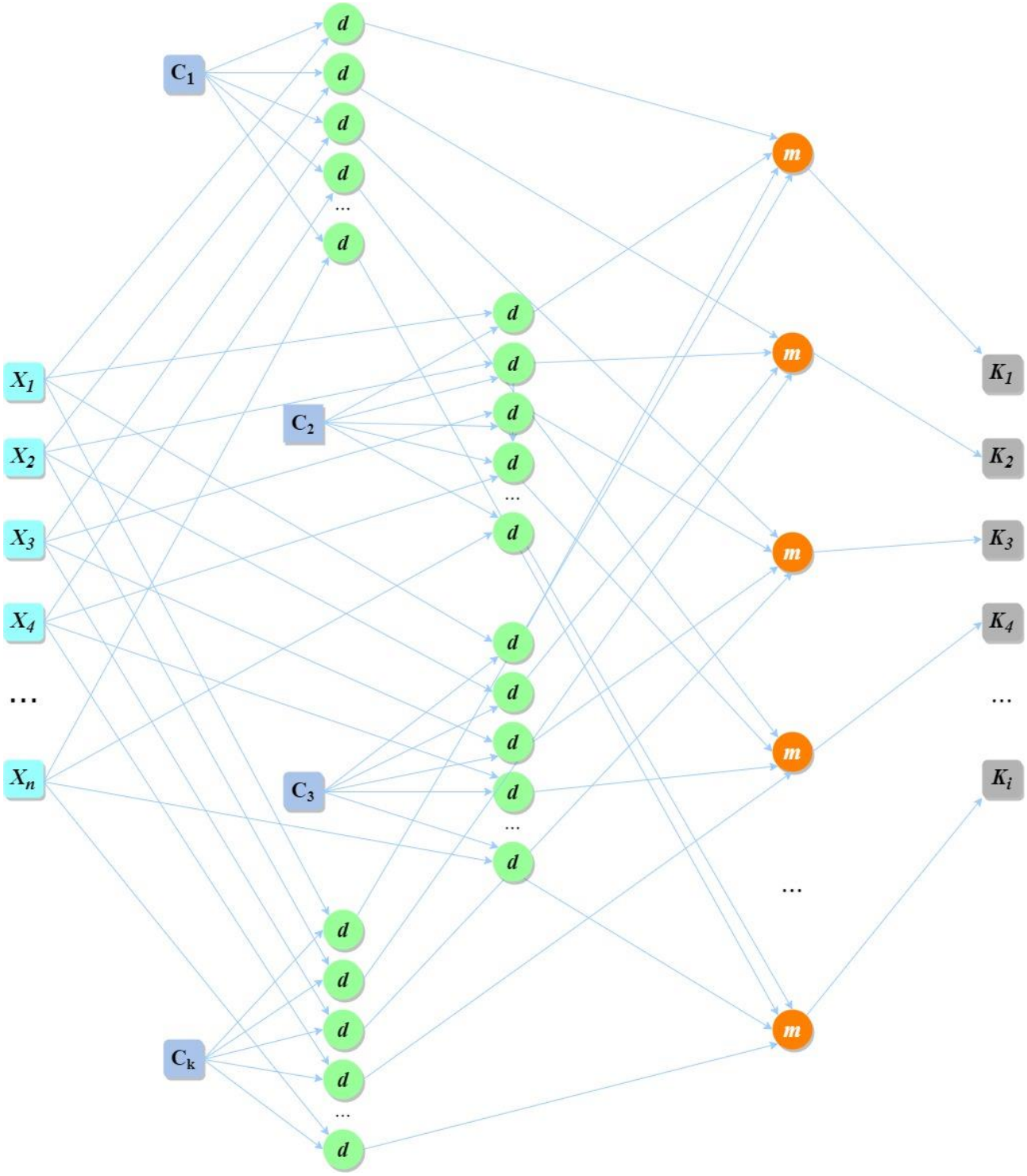


Рис. 2.5. Схема розподілу векторів по кластерах, де *d* - обчислення відстані між векторами та центрами кластерів; *m* - обчислення мінімуму

Кожна пара векторів даних x_i, де $i = 1,..,n$ і центрів кластера C_j, де $j = 1,..,k$: (x_i, C_j) подаються на незалежні вузли обчислення відстані між векторами *"d"*. Далі вузли обчислення відстані *"d"*, що відповідають одному і тому ж вихідному вектору x_i передаються на один вузол *"m"*, де виконується обчислення нової мітки кластера для кожного вектора x_i (з мінімальним результатом обчислення відстані). Врезультаті, видаються мітки кластерів $K_1,..,K_n$, такі що $\forall x_i,\ i = 1,..,n,\ x_i \in S_j \Leftrightarrow K_i = j$ [140].

Детальна схема обчислення відстаней між векторами та центрами мас кластерів представлена на рисунку 2.6.

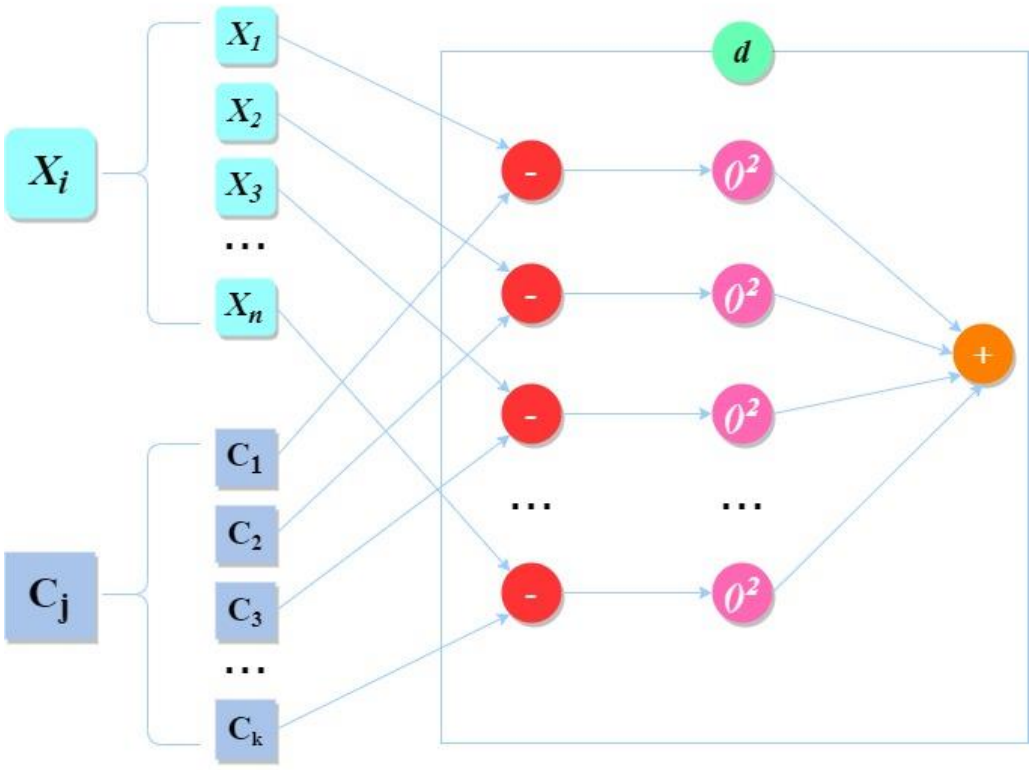


Рис. 2.6. Схема обрахунку відстані між векторами X_i, C_j

Як показано на рисунку 2.6., обчислення відстані між векторами $x_i = x_{i1},..,x_{in},\ C_j = C_{j1},..,C_{jn}$ (вузол *"d"*), може бути представлено як обчислення різниці між кожною парою компонент $(x_{in}, C_{jk}),\ z,k = 1,..,d$ (вузол "-"). Потім виконується піднесення до квадрату для кожного вузла "-" (вузол "$()^2$") та підсумовування виходів всіх вузлів "$()^2$" (вузол "+").

Крок 3. Перерахунок центрів кластерів, що передбачає k обчислень центрів мас C_i множин S_i, $i = 1, \dots, k$. Під час виконання цього кроку, проводиться оновлення центрів кластера за формулою обчислення центрів мас $\forall i = 1, .., k$ (2.13):

$$C_i^{(t+1)} = \frac{1}{\left|S_i^{(t)}\right|} \sum_{x_j \in S_i^{(t)}} x_j \qquad (2.13)$$

Перерахунок центрів кластерів графічно зображено на рисунку 2.7.

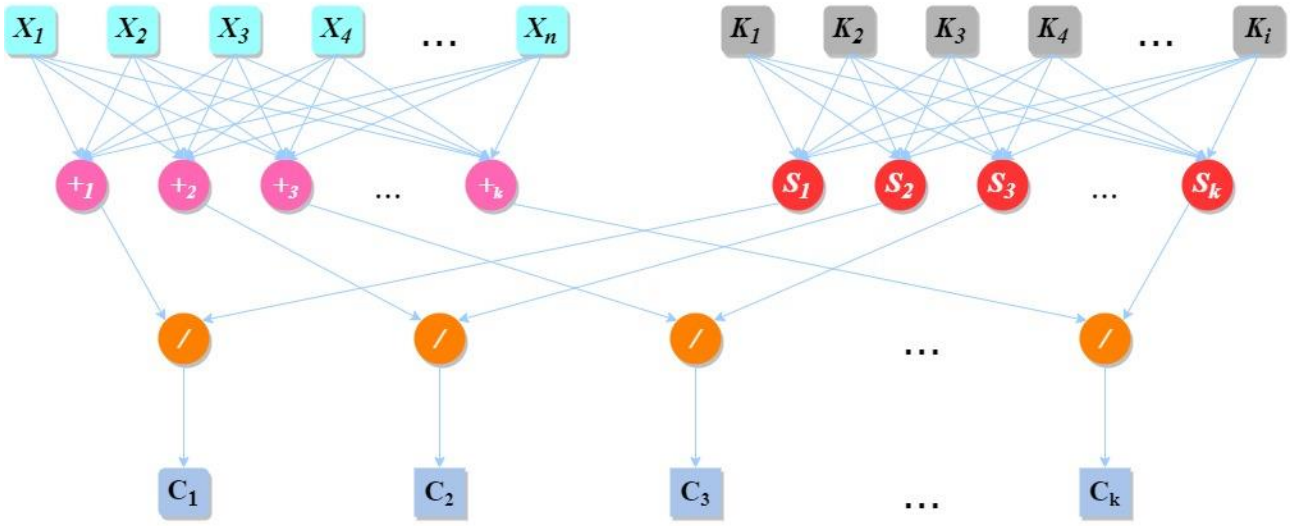


Рис. 2.7. Схема перерахунку центрів кластерів

На рисунку 2.7. зображені всі вектори $x_1, \dots, x_n$, що подаються у вузоли "$+_m$", де кожен вузол $+_m$, $m = 1, .., k$, відповідає операції додавання векторів кластера з номером m. Мітки кластера $K, \dots, K_n$ також передаються на вузли S_m, $m = 1, .., k$, де на кожному з яких обчислюється кількість векторів у відповідному кластері. Потім кожна пара виходів вузлів "$+_m$" і "S_m" подається на вузол "/", де проводиться розподіл суми векторів кластера на кількість елементів в ньому. Значення, обчислені на вузлах "/", присвоюються новим центрам кластерів.

Крок 4. Перерозподіл векторів даних моніторингу забруднення атмосферного повітря за допомогою обрахунку відстаней між векторами та оновленими центрами кластерів.

Крок 5. Якщо на попередньому кроці жоден вектор концентрацій параметрів забруднення атмосферного повітря не перейшов в інший кластер або

якщо виконано максимально допустиме число ітерацій, тоді перерозподіл кластерів припиняється [141].

Загальний алгоритм виконання етапу виділення кластерів тренувальної та тестової вибірок даних моніторингу зображено на рисунку 2.8.

Рис. 2.8. Схема алгоритму виконання кластерного аналізу для вибірок даних

З рисунку 2.8. випливає, що кластеризація векторів тренувальної та тестової вибірок даних моніторингу АП включає в себе наступні процедури:

Процедура 1. Виконання кластеризації методом k-means навчальної матриці по вхідних векторах ПЗб атмосферного повітря.

Процедура 2. Знаходження Евклідових відстаней між векторами концентрацій ПЗб атмосферного повітря тестової вибірки та векторами центрів мас кластерів навчальної матриці за формулою (2.6).

Процедура 3. Кластеризація всіх векторів тестової вибірки за допомогою віднесення кожного вектора до того кластеру, відстань до вектору центрів мас якого є мінімальною.

Власне реалізація методу введення додаткових вхідних компонентів векторів параметрів забруднення атмосфреного повітря на основі попереднього виділення кластерів описано в наступному підпараграфі 2.1.5.

2.1.5. Розширення входів тренувальної та тестової вибірок даних моніторингу повітряного середовища

Останнім етапом методу введення додаткових ознак шляхом попереднього виділення компактних множин точок є власне розширення входів векторів концентрацій тренувальної та тестової вибірок даних моніторингу атмосферного повітря.

Отже, після відкинення аномальних викидів та визначення приналежності всіх векторів концентрацій параметрів забруднення повітряного середовища тренувальної та тестової вибірок до встановленої кількості оптимальних кластерів, виконується розширення обох вибірок даних моніторингу атмосферного повітря.

Етап розширення вхідних векторів концентрацій параметрів забруднення АП виконується наступним чином:

Крок 1. До входів кожного вектора тренувальної вибірки даних моніторингу повітряного середовища додається k додаткових входів, де k – кількість попередньо виділених кластерів.

Крок 2. Аналогічно до входів кожного вектора тестової вибірки векторів концентрацій ПЗб атмосферного повітря додається k додаткових входів, де k – кількість попередньо виділених кластерів.

Крок 3. Якщо вектор концентрацій параметрів забрудення АП з треунувальної чи тестової вибірки належить до k-го кластера, то k-тій додатковій вхідній ознаці вектора присвоюється значення 1.

Крок 4. Решта додаткових вхідних ознак вектора концентрацій параметру забруднення заповнюються нулями, оскільки вектор належить лише до одного кластера.

Результатом виконання описаного методу розширення входів є тренувальна та тестова вибірки векторів концентрацій виду $x_{i1}, \ldots, x_{in}, 0_i, \ldots, 1_{ik}, \ldots, 0_i$, що зображено на рисунку 2.9.

X_1	X_2	…	X_{10}	X_{11}	Розширення кластерами																				Y
0,667	0,228	…	0,551	0,35	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2,6
0,624	0,125	…	0,672	0,36	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1,6
0,557	0,063	…	0,676	0,35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
0,536	0,044	…	0,673	0,35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0,9
0,523	0,021	…	0,634	0,34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0,7
0,561	0,061	…	0,672	0,34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1,1
0,662	0,182	…	0,683	0,35	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2,2
0,604	0,121	…	0,658	0,35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1,7
0,578	0,096	…	0,653	0,34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1,5
0,606	0,1	…	0,753	0,36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1,6
0,63	0,14	…	0,861	0,38	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1,9
0,642	0,169	…	0,9	0,4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	2,2
0,775	0,399	…	0,724	0,37	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	4,8
0,871	0,526	…	0,781	0,38	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	6,9
0,804	0,461	…	0,764	0,37	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	6,1
0,644	0,246	…	0,722	0,34	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3,9
0,473	0,09	…	0,715	0,32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1,5
0,448	0,05	…	0,685	0,31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
0,529	0,113	…	0,66	0,3	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1,7
0,512	0,123	…	0,673	0,29	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1,9
0,484	0,079	…	0,697	0,29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1,4
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

Рис. 2.9. Тренувальна вибірка даних розширена методом попереднього виділення кластерів

На рисунку 2.9. показано виконання етапу розширення входів на основі попереднього виділення КМТ для тренувальної вибірки концентрацій параметрів забруднення у даних моніторингу АП. Для вибірки даних виконано поділ на 20 компактних множити точок, що визначені під час вибору оптимальної кількості кластерів (параграф 2.1.3.).

Для того, щоб дати оцінку ефективності вперше розробленого методу введення додаткових атрибутів у вхідні вектори, шляхом встановлення їх належності до попередньо виділених кластерів, потрібно виконати його апробацію. Для цього розроблений метод застосовується для реалізації різних методів заповнення пропущених концентрацій параметрів забруднення у даних моніторингу атмосферного повітря.

2.2. Метод функційного розширення вхідних ознак векторів даних для НС МПГП

Для моделювання поверхонь відгуку нелінійного типу, які є характерними для задач прогнозування параметрів забруднення атмосферного повітря, найчастіше обирають багатошаровий перцептрон чи радіально-базисні мережі. Також можна обрати нейроподібну структуру моделі послідовних геометричних перетворень (НС МПГП) нелінійного типу.

У параграфі 2.1. розглянуто та описано метод введення додаткових атрибутів у вхідні вектори, шляхом встановлення їх належності до попередньо виділених кластерів, що базується на лінеаризації поверхні відгуку. У цьому ж параграфі розглянемо один з варіантів введення додаткових ознак – метод нелінійного розширення вхідних векторів даних на основі використання нелінійних функцій.

2.2.1. Основи методу нелінійного розширення входів Йох-Хан Пао

Даний метод вперше запропонований відомим вченим Йох-Хан Пао в науковій роботі [142]. Аналіз згаданого методу розширення входів показав, що введення додаткових функційних нелінійних вхідних ознак виконується шляхом розрахунку тригонометричних та інших нелінійних функцій. Наприклад, графік функцій синуса та косинуса є синусоїдою, що задається рівнянням (2.13):

$$v(t) = A.cos(2\pi ft + \varphi) \tag{2.13}$$

де A – ампілтуда сигналу;

$2\pi ft + \varphi$ – аргумент чи фаза функції, вираженої в радіанах;

f – частота сигналу, виражена в Герцах;

φ – початок фази (при $t=0$), виражений в радіанах;

t – початковий вхід тренувальної чи тестової вибірки даних моніторингу атмосферного потвіря.

Візуальний приклад графіку функції для введення додаткової ознаки за рахунок використання періодичності синусоїди зображено на рисунку 2.10., встановлення параметрів якої проілюстровано на рисунку 2.11. [142].

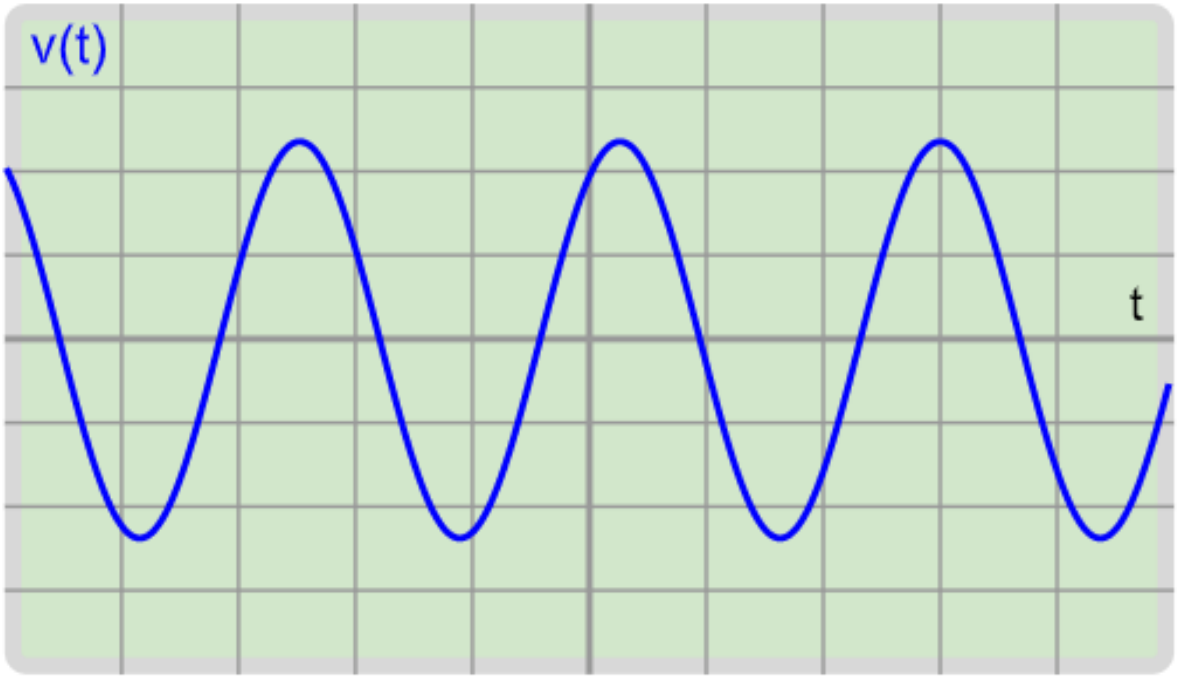


Рис. 2.10. Графік періодичної синусоїдної функції

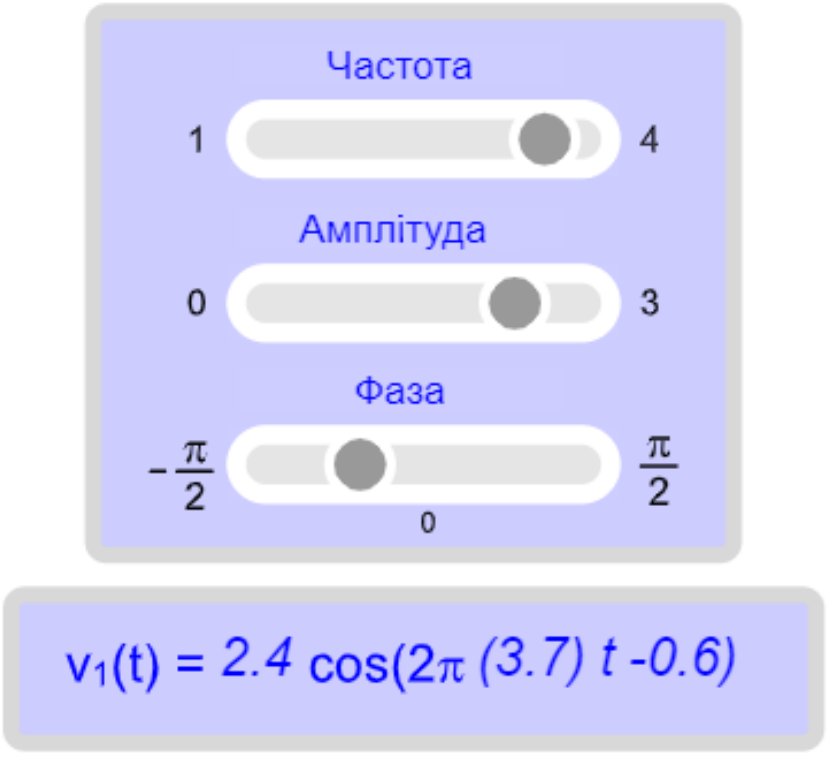


Рис. 2.11. Приклад налаштування параметрів синусоїдної функції

Застосування запропонованих Йох-Хан Пао нелінійних функцій типу синусоїдальних/косинусоїдальних не відповідає характеру нелінійності поверхні відгуку для завдань заповнення пропущених концентрацій параметрів забруднення у даних моніторингу атмосферного повітря. Тому пропонується удосконалення методу функційного розширення входів за допомогою обрахунку раціональних дробів, що забезпечує побудову нелінійних поверхонь в області вузлів інтерполяції та одночасно захищає від можливих викидів за їх межами.

Для апробації удосконалюваного методу функційного розширення вхідних ознак вибираємо використання нейроподіної структури моделі послідовних геометричних перетворень (НС МПГП) лінійного типу, де здійснюється апроксимація поверхні відгуку у вигляді гіперплощини.

Також, для оцінки ефективності удосконалюваного методу функційного введення додаткових ознак у вхідні вектори концентрацій параметрів забруднення повітряного середовища важливим є його реалізація під час застосування не лише НС МПГП лінійного та нелінійного типів, а й інших моделей машинного навчання.

В першому розділі запропоновано апробацію цього методу, використовуючи наступні проаналізовані нейроподібні та нейромережеві методи регресійного аналізу: лінійна регресія із застосуваням стохастичного градієнтного спуску (SGDr), дерево рішень, метод опорних векторів (SVR), адаптивний бустинг (AdaBoost) та НС МПГП. Тому важливим є дослідження ефективності виконання завдання заповнення пропусків у даних моніторингу атмосферного повітря шляхом застосування удосконалюваного методу функційного розширення входів на векторах концентрацій параметрів забруднення повітряного середовища.

Отже, як і розроблений метод розширення вхідних параметрів забруднення атмосферного повітря за допомогою попередньої кластеризації вхідних векторів даних, удосконалюваний нелінійний метод розширення входів шляхом застосування раціональних дробів складається з набору деяких процедур, що описані в наступному підпараграфі 2.2.2.

2.2.2. Удосконалення методу функційного розширення входів Йох-Хан Пао шляхом введення раціональних дробів

Удосконалення методу функційного розширення входів Йох-Хан Пао шляхом застосування раціональних дробів передбачає введення додаткових ознак у вхідні вектори даних за допогою обрахунку обернено-пропорційних квадратичних функцій. Реалізація методу функційного розширення входів складається з наступних етапів:

Етап 1. Розподіл вибірки даних на тренувальну та тестову.

Етап 2. Передобробка обох вибірок векторів.

Етап 3. Функційне розширення входів.

Етап 4. Оцінка ефективності удосконаленого шляхом застосування на різних моделей машинного навчання.

Метод функційного розширення входів Йох-Хан Пао удосконалюється за рахунок заміни нелінійних функцій типу синусоїдальних/косинусоїдальних на раціональні дроби. Таким чином, в основі нелінійного методу введення додаткових ознак у вхідні вектори даних складається з сукупності процедур, за допомогою яких можна розширювати дані для виконання завдання заповнення пропущених концентрацій параметрів забруднення атмосферного повітря. Ця сукупність включає в себе такі процедури:

Процедура 1. Відкинення аномальних викидів концентрацій параметрів забруднення у даних моніторингу атмосферного повітря. Для цього потрібно виконати такі пункти:

- підбір методу пошуку аномалій;
- власне виконання пошуку аномалій та їх відкинення.

Процедура 2. Виконання масштабування тренувальної та тестової вибірок векторів концентрацій ПЗб атмосферного повітря.

Процедура 3. Розширення вхідних векторів концентрацій параметрів забруднення тренувальної вибірки даних моніторингу повітряного середовища за допомогою обрахунку обернено-пропорційних квадратичних функцій. Введення

додаткових входів відбувається наступним чином: до входів кожного вектора додаємо *c* додаткових входів за формулою (2.14):

$$c_i = \frac{x_i^2}{(1+x_i^2)}, \tag{2.14}$$

де x_i – початкові вхідні концентрації вектора параметрів забруднення атмосферного повітря.

Процедура 4. Розширення вхідних векторів концентрацій параметрів забруднення атмосферного повітря тестової вибірки даних за рахунок використання тієї ж формули (2.14) що і для тренувальнох вибірки.

Ілюстрацію розширеної тестової вибірки даних моніторингу повітряного середовища зображено на рисунку 2.12.

X1	*X2*		*X10*	*X11*	*Функційне розширення*										*Y*
0,643	0,148	…	0,548	0,325	0,707	0,979	0,843	0,995	0,826	0,915	0,745	0,866	0,912	0,769	0,905	2
0,698	0,141	…	0,621	0,336	0,672	0,98	0,848	0,991	0,834	0,875	0,746	0,842	0,928	0,722	0,899	2,2
0,685	0,144	…	0,69	0,352	0,68	0,98	0,845	0,985	0,846	0,859	0,739	0,809	0,938	0,678	0,89	2,2
0,596	0,074	…	0,68	0,351	0,738	0,995	0,897	0,996	0,786	0,908	0,785	0,872	0,936	0,684	0,89	1,2
0,59	0,057	…	0,653	0,34	0,742	0,997	0,911	0,998	0,754	0,939	0,8	0,919	0,935	0,701	0,896	1,2
0,524	0,025	…	0,668	0,329	0,785	0,999	0,941	0,999	0,684	0,975	0,827	0,965	0,943	0,692	0,902	0,7
0,664	0,126	…	0,66	0,332	0,694	0,984	0,858	0,985	0,835	0,878	0,755	0,835	0,94	0,697	0,901	2
0,683	0,181	…	0,932	0,391	0,682	0,968	0,821	0,978	0,871	0,847	0,704	0,851	0,966	0,535	0,867	2,9
0,643	0,13	…	0,818	0,383	0,707	0,983	0,855	0,981	0,843	0,841	0,737	0,87	0,951	0,599	0,872	2,2
0,689	0,176	…	0,777	0,366	0,678	0,97	0,825	0,97	0,866	0,833	0,706	0,834	0,95	0,624	0,882	2,9
0,918	0,512	…	0,58	0,335	0,543	0,792	0,654	0,943	0,944	0,758	0,554	0,632	0,917	0,748	0,899	6,6
0,637	0,151	…	0,737	0,34	0,711	0,978	0,841	0,981	0,876	0,876	0,749	0,788	0,953	0,648	0,896	2,7
0,487	0,041	…	0,731	0,312	0,808	0,998	0,926	0,998	0,765	0,955	0,814	0,901	0,964	0,652	0,911	1
0,548	0,046	…	0,776	0,308	0,769	0,998	0,922	0,999	0,768	0,962	0,815	0,898	0,972	0,624	0,913	1,2
0,665	0,185	…	0,823	0,311	0,693	0,967	0,818	0,967	0,882	0,908	0,716	0,812	0,977	0,596	0,912	2,7
0,723	0,22	…	0,389	0,317	0,657	0,954	0,797	0,971	0,895	0,778	0,716	0,761	0,844	0,869	0,909	3,6
0,601	0,135	…	0,4	0,319	0,735	0,982	0,852	0,993	0,847	0,897	0,762	0,894	0,85	0,862	0,908	2
0,65	0,16	…	0,403	0,314	0,703	0,975	0,834	0,99	0,867	0,855	0,746	0,841	0,855	0,86	0,91	2,5
0,529	0,058	…	0,725	0,382	0,781	0,997	0,911	0,999	0,802	0,949	0,789	0,877	0,933	0,655	0,873	1,2
0,575	0,066	…	0,721	0,382	0,751	0,996	0,904	0,995	0,814	0,928	0,78	0,869	0,932	0,658	0,873	1,4
0,615	0,1	…	0,69	0,386	0,726	0,99	0,877	0,993	0,855	0,929	0,753	0,837	0,923	0,678	0,87	1,6
0,663	0,135	…	0,61	0,391	0,694	0,982	0,852	0,989	0,878	0,904	0,732	0,803	0,897	0,729	0,868	2,2
0,626	0,129	…	0,333	0,337	0,718	0,984	0,857	0,994	0,849	0,888	0,757	0,883	0,793	0,9	0,898	1,9
0,736	0,192	…	0,422	0,372	0,649	0,965	0,815	0,987	0,884	0,861	0,718	0,794	0,825	0,849	0,879	2,5
0,617	0,116	…	0,629	0,41	0,724	0,987	0,866	0,994	0,871	0,902	0,748	0,797	0,893	0,717	0,856	1,8
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

Рис. 2.12. Тестова вибірка розширених векторів концентрацій ПЗб АП

Отже, результатом виконання удосконаленого методу функційного розширення входів Йох-Хан Пао є тренувальна та тестова вибірки даних із введеними додатковими ознаками векторів концентрацій ПЗб АП.

2.3. Порівняльна оцінка ефективності методів заповнення пропусків у даних моніторингу атмосферного повітря

2.3.1. Опис досліджуваної вибірки концентрацій параметрів забруднення повітряного середовища для завдання заповнення пропусків

Моделювання роботи досліджених методів заповнення пропусків у даних виконується на реальній задачі екологічного моніторингу. Реалізація цих методів відбувається на наборі даних, котрий містить 9000 спостережень погодинних усереднених відповідей з 5 металевих оксидних хімічних датчиків, вбудованих в хімічний мультисенсорний пристрій для визначення якості повітря. Частину переліку вимірів, що надані спільним контрольно-перевіреним аналізатором, наведено у таблиці 2.1.

Таблиця 2.1.

Уривок даних моніторингу атмосферного повітря за 2004-2005 роки з погодинним вимірюванням концентрацій параметрів забруднення

Дата	Час	CO	SnO_2	C_6H_6	Ti	NO	WO	NO_2	WO_2	InO	T	RH	AH
20.02.2005	19:00:00	1,1	1016	2,9	647	154	906	119	997	649	7,1	62,6	0,6341
20.02.2005	20:00:00	1,5	1076	4,6	745	221	801	149	1040	841	7,4	59,3	0,6123
20.02.2005	21:00:00	1,4	1029	3,8	699	193	851	140	1001	795	7,7	56,9	0,5994
20.02.2005	22:00:00	1,1	984	3,1	661	130	898	105	965	638	7,7	54,4	0,5753
20.02.2005	23:00:00	1,3	1003	3,7	694	156	876	116	976	663	7,7	53,0	0,5572
21.02.2005	0:00:00	1,4	1009	3,3	671	148	877	N/A	1010	610	6,7	62,8	0,6188
21.02.2005	1:00:00	1,1	946	2,1	588	93	1005	80	948	480	6,6	61,9	0,6056
21.02.2005	6:00:00	0,6	949	2,1	589	77	987	63	1001	464	3,3	84,3	0,6617
21.02.2005	7:00:00	1,2	1054	5,5	786	183	766	123	1129	699	4,2	80,9	0,6742
21.02.2005	8:00:00	2,2	1214	10,9	1010	256	609	164	1282	1006	5,1	75,0	0,6646
21.02.2005	9:00:00	2,9	1253	13,3	1095	355	566	218	1375	1206	4,6	77,6	0,6651
21.02.2005	10:00:00	2,9	1197	9,6	964	491	614	254	1260	1122	3,8	84,3	0,6811
21.02.2005	11:00:00	2,0	1156	7,0	859	387	684	210	1196	1069	6,2	77,4	0,7383
21.02.2005	12:00:00	1,7	1132	6,6	837	384	694	208	1199	1062	7,7	67,6	0,7133
21.02.2005	13:00:00	2,2	1207	9,3	950	368	623	204	1303	1162	8,1	64,8	0,7039
21.02.2005	14:00:00	1,8	1127	7,9	896	258	664	172	1227	1010	8,1	64,0	0,6942
21.02.2005	15:00:00	1,6	1112	6,9	851	231	694	157	1184	931	6,9	71,3	0,7118
21.02.2005	16:00:00	1,9	1186	8,6	925	290	641	181	1272	1025	7,0	73,7	0,7425
21.02.2005	17:00:00	2,2	1192	10,0	976	335	618	194	1290	1110	7,3	70,0	0,7195
21.02.2005	18:00:00	2,2	1177	8,8	931	293	656	193	1281	979	5,8	77,8	0,7223

Продовження Таблиці 2.1.

21.02.2005	19:00:00	4,9	1478	20,2	1301	526	469	252	1682	1409	5,4	80,9	0,7293
21.02.2005	20:00:00	6,3	1486	19,7	1289	607	466	283	1633	1438	5,5	79,8	0,7261
21.02.2005	21:00:00	2,7	1118	7,1	863	326	680	208	1194	1083	5,6	74,0	0,6808
21.02.2005	22:00:00	1,4	1019	4,4	730	188	811	143	1082	916	5,6	71,9	0,6604
21.02.2005	23:00:00	N/A	961	2,9	643	163	914	124	1046	825	6,4	70,7	0,6848
22.02.2005	0:00:00	0,8	931	2,1	591	N/A	1015	89	1015	704	6,3	70,1	0,6733
…	…	…	…	…	…	…	…	…	…	…	…	…	…

Прилади знаходилися на полі у значно забрудненій місцевості, на рівні дороги. Набір даних, складається з погодинних усереднених концентрацій оксиду вуглецю (CO), не метанових вуглеводнів (SnO_2), бензолу (C_6H_6), титану (Ti), загальних оксидів азоту (NO) та діоксиду азоту (NO_2), оксидів вольфраму (WO) та діоксиду вольфраму (WO_2), оксиду індію (InO), температури повітря (T), відносної (RH) та абсолютної (AH) вологості повітря. Відсутні значення, котрі потрібно відновити, позначаються значенням «*N/A*». Докази перехресної чутливості, а також концептуальні та датчикові дрейфи присутні та описані в [143], що в кінцевому підсумку впливає на можливості оцінки концентрації датчиків.

У дослідженні, на противагу існуючим методам заповнення пропусків, пропонуються два методи заповнення пропусків у даних моніторингу забруднення атмосферного повітря: лінійний та нелінійний методи, побудовані із застосуванням нейромережевого підходу. Згідно з ним, спочатку здійснюється відповідне навчання обраної ШНМ а в режимі застосування отримується передбачення пропущених даних. Оскільки пропущені дані можуть бути присутніми для різних параметрів у моніторингу стану забруднення повітряного середовища, виникає потреба навчання великої кількості НМ в різних режимах застосування. Так як застосування НМ широковідомих архітектур типу одношарових та багатошарових перцептронів пов’язане зі значними затратами часу на налагодження їх параметрів [15], тому обрано нейронні структури моделі послідовних геометричних перетворень (НС МПГП), які навчаються та функціонують повністю в автоматичному режимі.

2.3.2. Результати порівняння методів заповнення пропущених концентрацій параметів забруднення атмосферного повітря

Для перевірки ефективності розроблених методів необхідно реалізувати регресійні методи, вибір яких обгрунтовано в попередньому розділі, та виконати заповнення пропущених параметрів забруднення атмосферного повітря на початкових експериментальних даних. Для того, щоб мати можливість визначити точність заповнення пропусків, першим кроком, котрий описаний раніше, є видалення пропущених рядків у даних. Наступним кроком є розділення векторів концентрацій ПЗб АП випадковим чином на тренувальну і тестову вибірки даних, та виконання заповнення пропусків різними методами.

2.3.2.1. Результати заповнення пропущених концентацій оксиду карбону

Таким чином, у роботі виконано порівняння ефективності вибраних регресійних методів заповнення пропущених концентрацій у даних моніторингу атмосферного повітря. Заповнення пропусків виконано для таких параметрів забруднення як оксиду карбону (CO) та діоксиду азоту (NO_2). Результати навчання та застосування регресійних методів заповнення пропущених концентрацій оксиду карбону (вуглекислого газу) наведено у таблиці 2.2.

Таблиця 2.2.

Похибки заповнення пропущених концентрацій вуглекислого газу

CO	Методи / Похибки	**MAE, train**	**MAPE, train**	**RMSE, train**	**RMSE_M, train**
	SGDr	0,382741394	31,83880886	0,558597502	5,476446098
	SVR	0,306195453	26,70798562	0,450737225	4,418992405
	Adaptive Boosting	0,42788379	43,16993034	0,543653981	5,329940987
	НС МПГП	0,279766125	23,00416982	0,438360743	4,297654345
	Методи / Похибки	**MAE, test**	**MAPE, test**	**RMSE, test**	**RMSE_M, test**
	SGDr	0,406174453	30,68429514	0,60432209	5,078336892
	SVR	0,329298561	25,57425911	0,497646905	4,181906761
	Adaptive Boosting	0,45938908	42,15117683	0,606752857	5,098763506
	НС МПГП	0,305470114	22,38388325	0,490839372	4,124700607

Графічне представлення результатів навчання та застосування досліджених методів заповнення пропущених концентрацій вуглекислого газу зображено на рисунках 2.13а. - 2.13г.

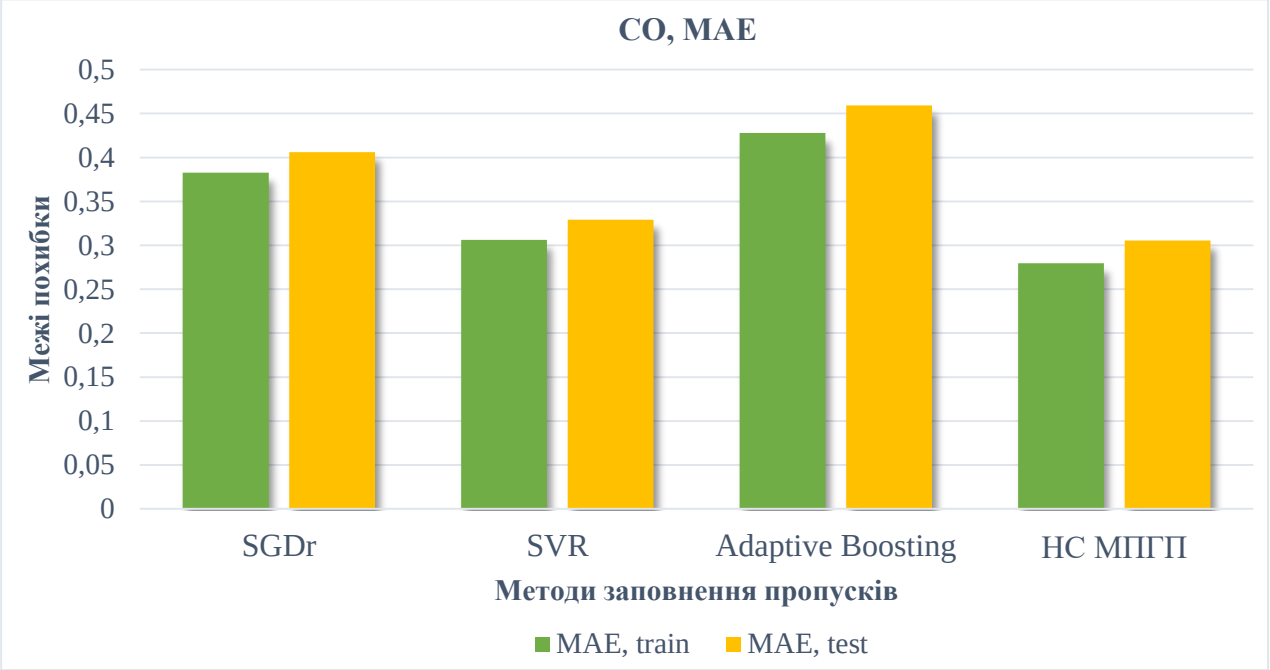


Рис. 2.13а. Похибки заповнення пропусків оксиду карбону, MAE

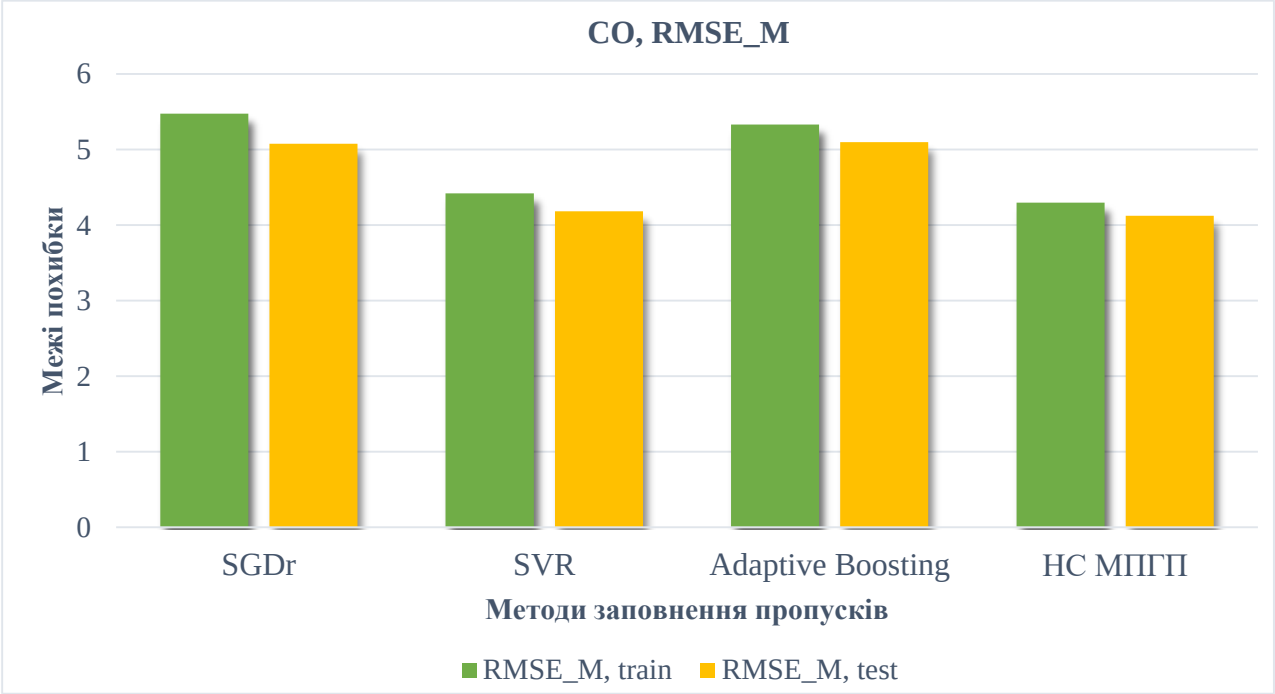


Рис. 2.13б. Похибки заповнення пропусків оксиду карбону, RMSE_M

Також встановлено похибки навчання та заповнення пропущених концентрацій оксиду карбону методом середнього значення (рис. 2.13в. та 2.13г).

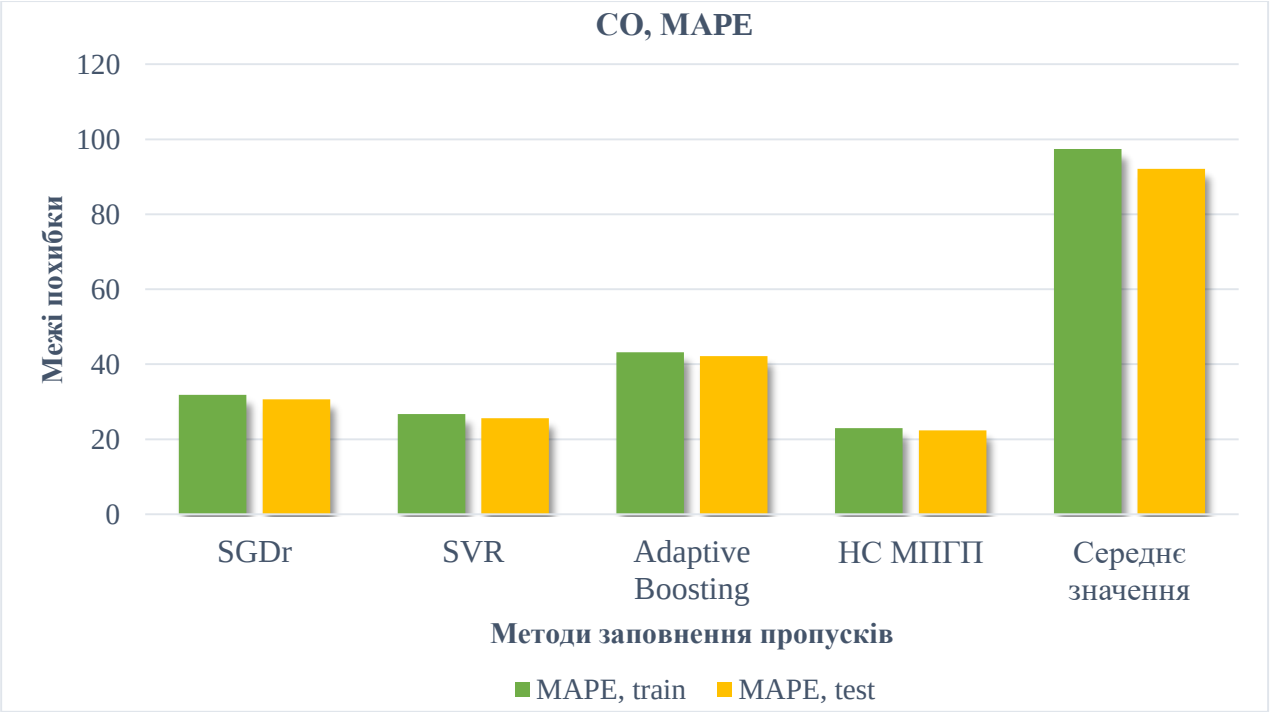


Рис. 2.13в. Похибки заповнення пропусків оксиду карбону, MAPE

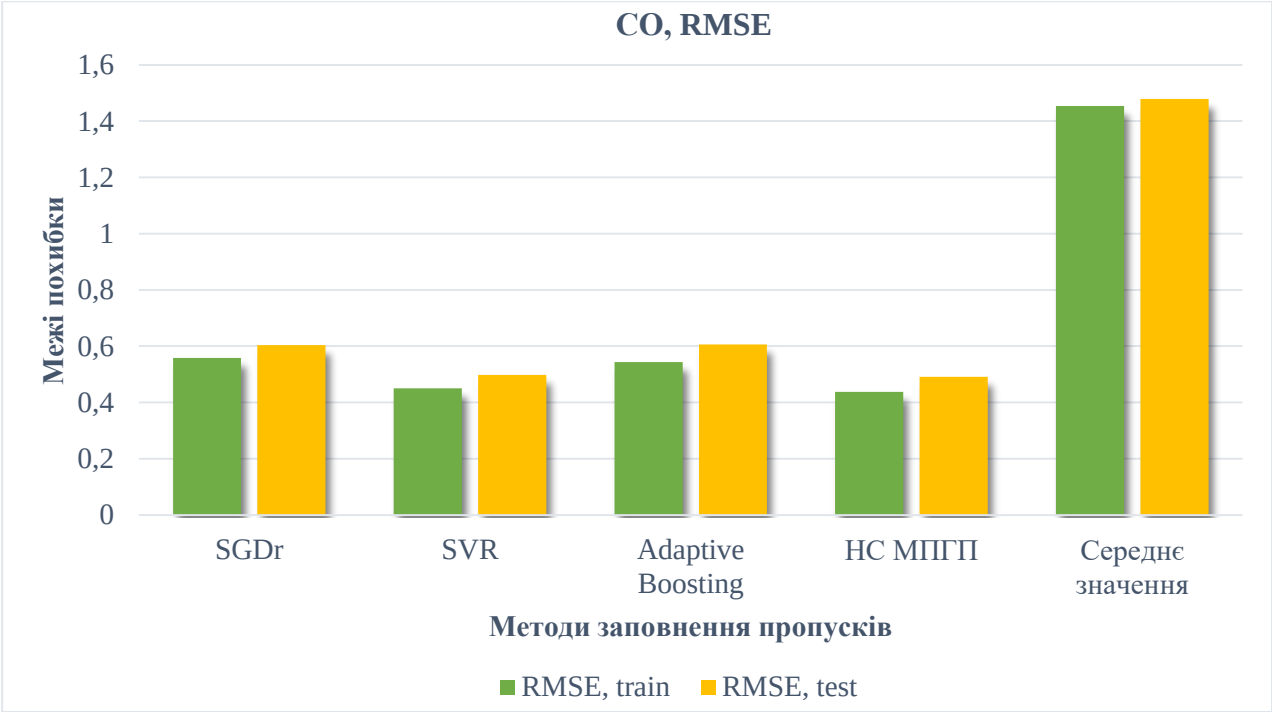


Рис. 2.13г. Похибки заповнення пропусків оксиду карбону, RMSE

Отже, з таблиці 2.2. та рисунків 2.13а. – 2.13г. випливає, що використання нейропобідних структур МПГП є найефективнішим методом заповнення пропущених концентрацій вуглекислого газу серед інших методів, оскільки результує з найменшими похибками.

2.3.2.2. Результати заповнення пропущених концентацій діоксиду азоту

Навчання та застосування регресійних методів виконується також для завдання заповнення пропущених концентрацій діоксиду азоту. Результати використання досліджуваних методів наведено у таблиці 2.3.

Таблиця 2.3.

Похибки заповнення пропущених концентрацій діоксиду азоту

	Методи / Похибки	**MAE, train**	**MAPE, train**	**RMSE, train**	**RMSE_M, train**
NO_2	**SGDr**	21,50356395	23,51554899	27,8767928	8,381477088
	SVR	17,4649983	18,29066423	22,61915912	6,900047841
	Adaptive Boosting	18,53920991	21,53884482	23,28740258	7,001624348
	НС МПГП	17,13095403	17,97542875	22,85697504	6,872211378
	Методи / Похибки	**MAE, test**	**MAPE, test**	**RMSE, test**	**RMSE_M, test**
	SGDr	21,90005002	26,62655368	28,65130623	9,239376404
	SVR	18,04394409	22,22201092	24,67745703	7,812949704
	Adaptive Boosting	19,21834643	24,55731486	24,76727793	7,986868084
	НС МПГП	17,98804174	21,40548137	24,03325785	7,750163769

Результати навчання та застосування регресійних методів заповнення пропущених концентрацій вуглекислого газу наведено на рисунках 2.14а. - 2.14г.

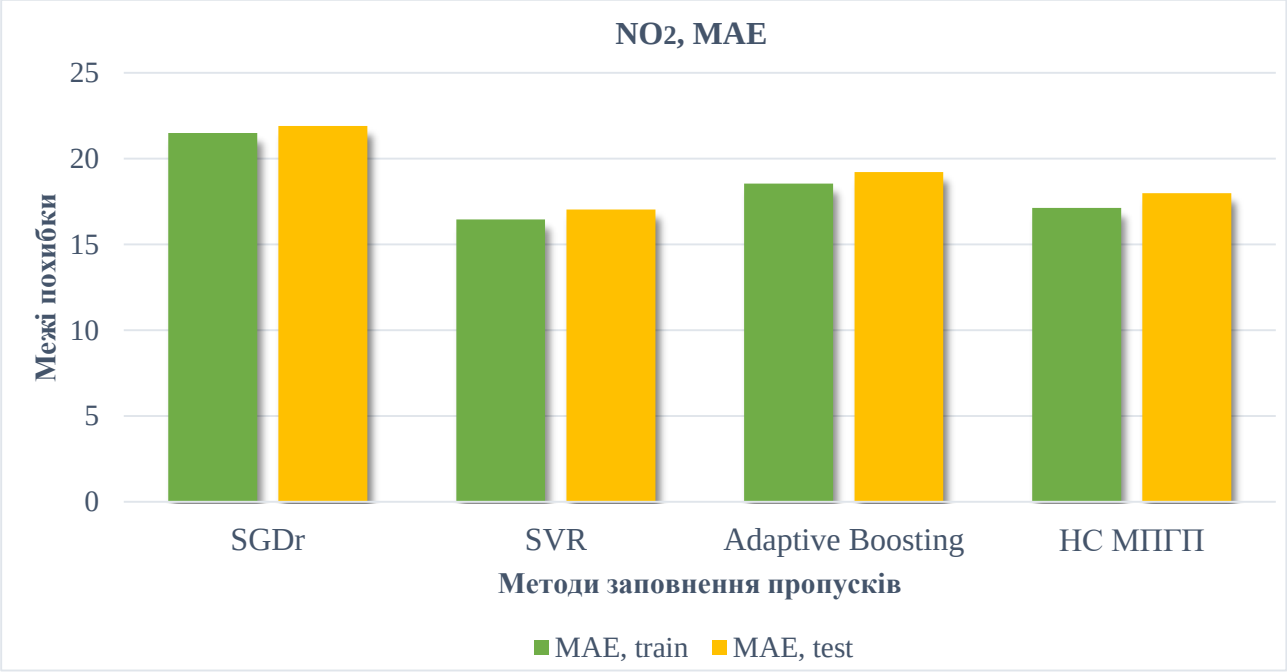


Рис. 2.14а. Похибки заповнення пропусків діоксиду азоту, MAE

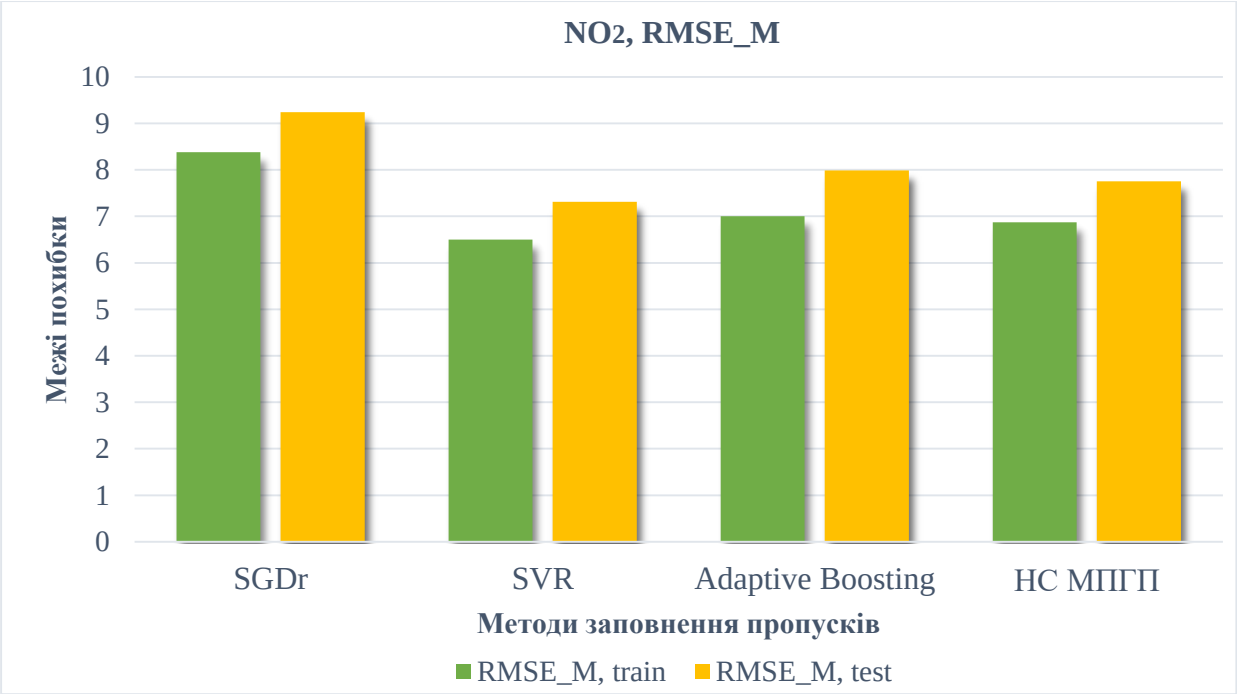


Рис. 2.14б. Похибки заповнення пропусків діоксиду азоту, RMSE_M

Додатково до регресійних методів, досліджено заповення пропущених концентрацій діоксиду азоту методом середнього значення. Встановлено, що похибки навчання та застосування (MAPE та RMSE) методу середнього значення є гіршими за решту похибок знайдених шляхо використання регресійних методів (рис. 2.14в. та 2.14г).

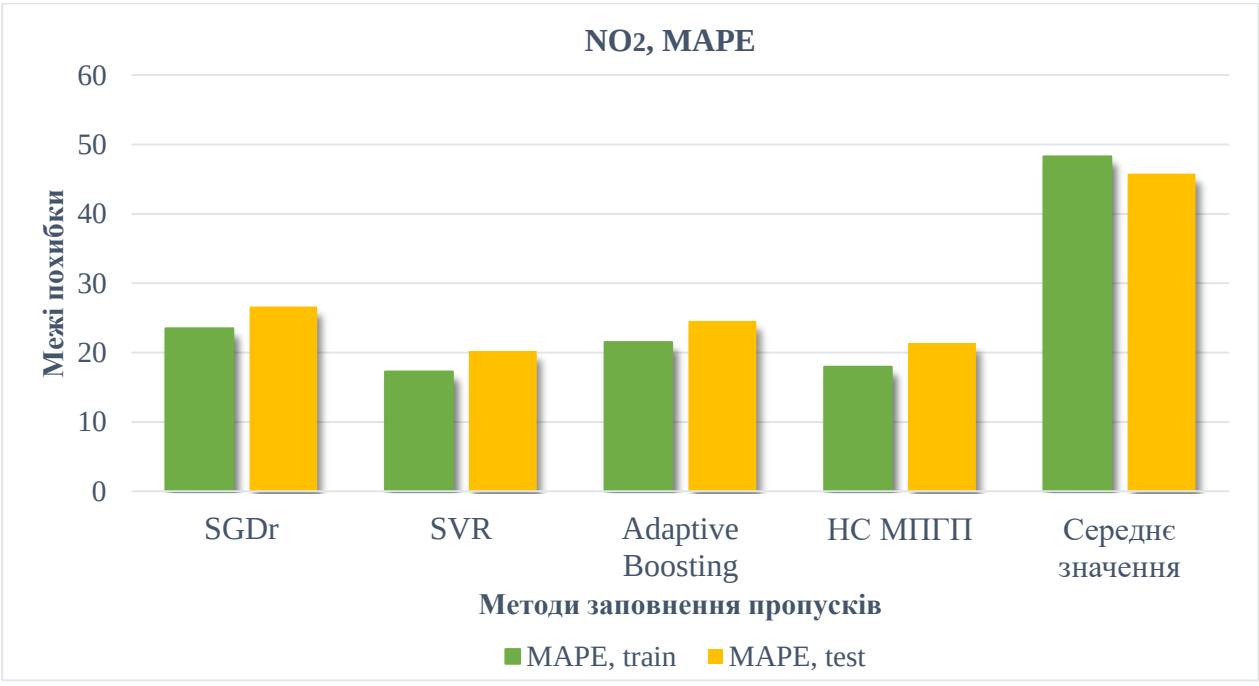


Рис. 2.14в. Похибки заповнення пропусків діоксиду азоту, MAPE

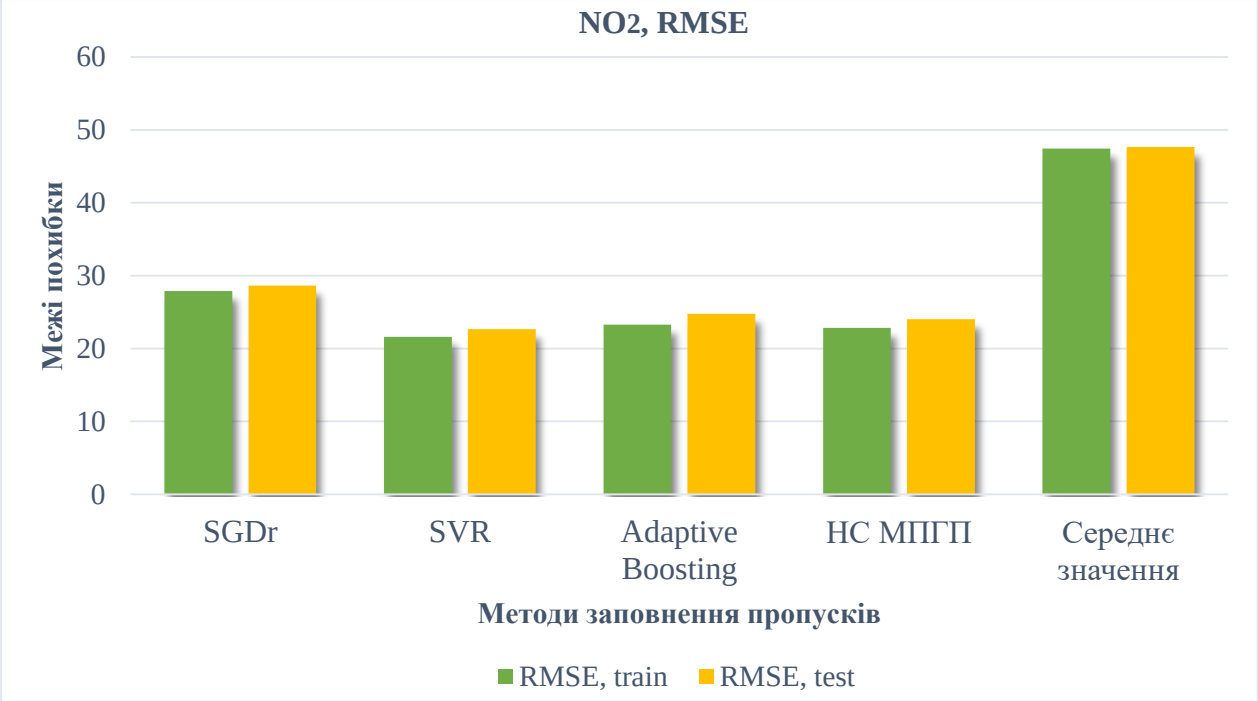


Рис. 2.14г. Похибки заповнення пропусків діоксиду азоту, RMSE

Під час виконання порівняльного аналізу наведених результатів заповнення пропущених концентрацій вуглекислого газу у таблиці 2.3. та рисунках 2.14а. – 2.14г. випливає, що використання НС МПГП є найефективнішим методом серед інших методів, оскільки результатами його навчання та застосування є найменші похибки серед інших досліджених методів.

Доведено, що похибки MAPE під час навчання та застосування НС МПГП для заповнення пропущених концентрацій становлять 23 % та 22,38 % для вуглекислого газу, і 17,98 % та 21,41 % для діоксиду азоту. Таким чином, середня похибки навчання НС МПГП становить 20,49 %, а а середня похибка заповнення пропущених ПЗб АП становить 21,9 %, відповідно.

Для підвищення точності заповнення пропущених концентрацій ПЗб АП розроблено та удосконалено методи розширення вхідних ознак даних. Важливим є реалізація описаних у цьому розділі методів введення додаткових вхідних ознак векторів даних під час застосування досліджуваних регресійних методів заповнення пропусків, а саме методу на основі НС МПГП, для оцінки ефективності розробленого та удовконаленого методів підвищення точності.

ВИСНОВКИ ДО РОЗДІЛУ 2

1. Оскільки розподіл векторів реалізацій відповідає положенням гіпотези компактності, у розділі розроблено та удосконалено методи введення додаткових вхідних ознак векторів даних для підвищення точності заповнення пропусків, які базуються на основі теореми Ковера про розділення образів.

2. Розроблено метод розширення вхідних ознак векторів даних на основі попереднього виділення компактних множин точок. Для цього виконано аналіз методів кластеризації та обгрунтовано вибір методу k-means. Також, детально описано етапи реалізації розробленого методу, серед яких вказані: пошук та відкинення аномальних концентрації ПЗб АП; вибір оптимальної кількості кластерів; етап кластеризації векторів концентрацій шкідливих домішок; та, власне, розширення вхідних ознак векторів даних моніторингу забруднення повітряного середовища.

3. Удосконалено метод функційного введення додаткових вхідних ознак векторів даних, що розроблений відомим вченим Йок-Хан Пао. Наведено основні принципи реалізації методу, що удосконалюється, та проаналізовані функції для розширення входів відомим методом. Описано етапи удосконалення методу Йок-Хан Пао за рахунок використання раціональних дробів.

4. Також виконано порівняльну оцінку ефективності методів заповнення пропусків у даних моніторингу забруднення атмосферного повітря та доведено, що найефективнішим є метод на основі НС МПГП, оскільки результує з точнішими результатами, ніж інші досліджувані методи. Встановлено, що середня відносна похибка заповнення пропущених концентрацій параметрів забруднення АП на основі НС МПГП становить 21,9 %.

5. Оскільки пріоритетним є покращення якості оцінки та прогнозування стану довкілля за рахунок підвищення точності заповнення пропущених коцентрацій ПЗб, у другому розділі описано реалізацію розробленого та удосконаленого методів розширення вхідних ознак векторів атрибутів шкідливих домішок для підвищення точності заповнення пропусків у даних моніторингу АП.

РОЗДІЛ 3. МЕТОДИ НЕЙРОМЕРЕЖЕВОГО ПРОГНОЗУВАННЯ ПАРАМЕТРІВ ЗАБРУДНЕННЯ ПОВІТРЯНОГО СЕРЕДОВИЩА

Теоретичною базою, що визначає концептуальну можливість прогнозування параметрів забруднення атмосферного повітря на підставі відомої передісторії є теорема Такенса [144]. Ця теорема заснована на операціях вкладення множин, що стверджує: якщо часовий ряд породжується динамічною системою, тобто значення x_t є довільною функцією системи, існує така глибина занурення d, яка забезпечує однозначне передбачення наступного значення часового ряду. Отже, прогнозування часового ряду зводиться до задачі інтерполяції функції багатьох змінних. Однак на практиці можливості здійснення прогнозів обмежуються доступною довжиною передісторії, а також наявними похибками збору параметрів забруднення атмосферного повітря для контролю стану навколишнього середовища.

Розглянемо ДС виду $\dot{x} = f(x)$ в припущенні, що через будь-яку точку фазового простору x проходить тільки одна траєкторія, і якщо в момент t_0 траєкторія виявилася в точці $x_0 = x(t_0)$, то в момент $t_0 + \tau$ вона виявиться в точці

$$x_1 = x(t_0 + \tau), \tag{3.1}$$

причому x_1 залежить від x_0 і τ, але не від t_0. Тобто існує відображення φ^τ, що переводить x_0 в x_1:

$$x_1 = \varphi^\tau(x_0). \tag{3.2}$$

Тоді члени оброблюваного тимчасового ряду (3.3):

$$h_i = h(t_i) = \Phi\big(x(t_i)\big) = \Phi(x_i) \tag{3.3}$$

можна уявити як

$$h_k = \Phi(x_k) = \Phi_0(x_k)$$

$$h_{k+1} = \Phi(x_{k+1}) = \Phi\big(\varphi^\tau(x_k)\big) = \Phi_1(x_k)$$

$$h_{k+2} = \Phi(x_{k+2}) = \Phi\big(\varphi^{\tau}(\varphi^{\tau}(x_k))\big) = \Phi_2(x_k) \qquad (3.4)$$

…

$$h_{k+m-1} = \Phi_{m-1}(x_k)$$

Тобто якщо ввести новий m-мірний вектор $z_k = (h_k, h_{k+1}, .., h_{k+m-1})$, то повинна існувати функція Λ, що залежить тільки від вихідної динамічної системи (тобто функцій f або φ) і параметрів m і τ, така що $z = \Lambda(\mathrm{x})$.

Виникає питання, чи не можна звернути функцію Λ, тобто виразити невідомий вектор x через відомий z. Аналітично це зробити не вдається, але Ф. Такенсоном була доведена теорема, яка стверджує, що майже для всіх τ і $m \geq 2d + 1$, де d - розмірність аттрактора, відображення $z = \Lambda(\mathrm{x})$ буде взаємно однозначним і безперервним. Якщо ж в просторі z-векторів виділити множину, на яку відображаються вектори x, то на цій множині відображення буде оборотним, і можна умовно записати $x = \Lambda^{-1}(\mathrm{z})$ [145].

Підставивши вираз для x в співвідношення $h_{k+m} = \Phi_m(x_k)$, ми отримаємо для всіх k:

$$h_{k+m} = \Phi_m(\Lambda^{-1}(\mathrm{z})) = F(z) = F(h_k, h_{k+1}, .., h_{k+m-1}) \qquad (3.5)$$

Завдання прогнозування параметрів забруднення атмосферного повітря, можна виконати за допомогою нейронних мереж завдяки відновленню невідомої функції по набору прикладів, заданих історією даного часового ряду. Згідно з теоремою Такенса, для деяких значень m існує залежність (3.6):

$$x_{t+m} = F(x_t, x_{t+1}, .., x_{t+m-1}) \qquad (3.6)$$

Таким чином, після завершення процесу навчання нейронна мережа імовірно "вміє" обчислювати функцію F і для продовження часового ряду потрібно подати на вхід мережі $h_{N-m+1}, .., h_N$ компоненти часового ряду і отримати на виході шукану величину - h_{N+1}.

Відома велика кількість різних методів прогнозування, що ґрунтуються на аналізі минулих значень часової послідовності, тобто методів, які використовують схему екстраполяції, коли властивості послідовності, виявлені на даному інтервалі часу, поширюються за його межі. При цьому передбачається, що властивості послідовності в майбутньому будуть такими ж, як в минулому і сьогоденні [146].

У дисертаційній роботі досліджено ряд методів нейромережевого прогнозування ПЗб атмосферного повітря, серед яких багатошаровий перцептрон (БШП), метод опорних векторів на основі регресії (SVR), лінійна регресія із застосуванням стохастичного градієнтного спуску (SGDr), адаптивний бустинг (AdaBoost), випадковий ліс (Random Forest) та дерево рішень, а також комітет НС різних типів [81 – 86, 101 – 113]. Важливим є застосування згаданих методів для виконання завдання прогнозування ПЗб АП, щоб оцінити найефективніший методу серед інших, взятих для порівняння.

Оскільки розвиток комп'ютерних технологій вимагає постійного покращення якості прогнозування, тому важливим є розробка нейроподібного методу підвищення точності прогнозування та розвинення методу прогнозування концентрацій параметрів забруднення атмосферного повітря за допомогою нейромережевої ідентифікації коефіцієнтів лінійних поліномів на етапі навчання.

Метод підвищення точності прогнозування ПЗб АП грунтується на виконанні лінійно-східчастої апроксимації за рахунок поєднання НС різних типів. Таке поєднання дозволяє скоригувати результати регресійного аналізу та зменшити похибку прогнозування. Розвинення пришвидшеного методу прогнозування параметрів забруднення атмосферного повітря в реальному часу виконується за рахунок ідентифікації коефіцієнтів лінійних поліномів під час навчання нейроподібної структури МПГП.

Відомо, що процес прогнозування в кожному конкретному випадку вимагає індивідуального підходу і зазвичай включає в себе цілий ряд процедур.

Для завдання прогнозування параметрів забруднення атмосферного повітря потрібно виконати наступні процедури:

Процедура 1. Аналіз часової послідовності на предмет наявності пропущених і випадаючих значень. Корекція цих значень.

Процедура 2. Перевірка послідовності на стаціонарність.

Процедура 3. Визначення наявності тренду і його типу. Визначення наявності періодичності в послідовності.

Процедура 4. Аналіз послідовності на предмет необхідності попередньої обробки (масштабування даних одним із загальновідомих методів приведення значень в деякий необхідний діапазон).

Процедура 5. Вибір способу та моделі прогнозування (навчання з вчителем чи без вчителя, розподіл на навчальну та тестову вибірки даних).

Процедура 6. Прогнозування на підставі обраної моделі (визначення та налаштування параметрів моделі).

Процедура 7. Оцінка точності прогнозу моделі (визначення похибок навчання та власне прогнозування).

Процедура 8. Визначення адекватності та виконання оцінки ефективності обраної моделі за рахунок порівняння з іншими моделями (методами / алгоритмами прогнозування).

Слід підкреслити той факт, що визначення та налаштування параметрів моделі в режимах навчання та застосування, і безпосередньо отримання прогнозу, є лише невеликою частиною загальної процедури виконання прогнозування.

Далі детально опишемо процедури та етапи реалізації розробленого нейроподібного лінійно-східчастого методу прогнозування ПЗб атмосферного повітря за домопомогою використання комітету нейроподібних структур різних типів для підвищення точності передбачення за рахунок корекції похибки у режимі застосування.

3.1. Метод короткотермінового прогнозування параметрів забруднення атмосферного повітря на основі комітету НС різних типів

Метод короктротермінового прогнозування ПЗб АП з розширеним горизонтом передбачення за рахунок корекції похибки на основі комітету НС різних типів має дещо видозмінені процедури, ніж загальний метод прогнознозування часових послідовностей. Процедури розроблюваного методу краще назвати етапами, оскільки вони включають в себе дадатковий перелік підпроцедур.

Перші *три етапи* пропонованого у роботі методу короткоретмінового прогнозування ПЗб повітряного середовища співпадають з загальним методом прогнозування часових послідовностей. Отже, розробка методу розпочинається з *етапу аналізу* ПЗб АП *на пропуски*: якщо у даних моніторингу довкілля наявні пропущені концентрації шкідливих домішок, тоді виконується заповнення пропусків; у випадку відсутності пропущених даних – відбувається перехід на *етап перевірки даних на стаціонарність*. Якщо атрибути вибірки є стаціонарними, тоді відбувається застосування методу часових вікон. Третім *етапом* є *виділення тренду*, де важливим є наявність вбудованого виділення тренду у НС МПГП.

Четвертий *етап* відрізняється від загальної методики прогнозування тим, що *масштабування даних* є обов'язковим, оскільки ПЗб АП знаходяться в різних числових діапазонах. Наступним є *етап розподілу* однієї *вибірки* даних на дві: *навчальну та тестову*, після чого відбувається застосування НС МПГП лінійного типу для прогнозування на один крок та багато кроків вперед.

Етап багатокрокового прогнозування виконується до тих пір, поки похибка прогнозування не починає значно рости (тобто відбувається пошук горизонту прогнозування). Останнім етапом реалізації методу короткотермінового прогнозування ПЗб АП є виконання корекції похибки за рахунок застосування нейроподібних структур різних типів.

Блок-схема описаних етапів розробленого нейропобідного методу прогнозування ПЗб атмосферного повітря зображена на рисунку 3.1.

Рис. 3.1. Алгоритм реалізації розробленого методу прогнозування ПЗб АП

Кожен з етапів реалізації методу нейроподібного лінійно-східчастого методу прогнозування ПЗб АП скаладається з набору деяких процедур та кроків, що описані в наступних параграфах.

3.1.1. Перевірка часової послідовності даних моніторингу атмосферного повітря на стаціонарність

Поняття екстраполяції передбачає, що деякий процес в майбутньому буде розвиватися так само як в минулому і сьогоденні [147]. Іншими словами, мова йде про стаціонарність процесу. Стаціонарний процес є найбільш привабливим з точки зору побудови різвих видів прогнозів, але, на жаль, в природі таких процесів не існує, оскільки будь-який реальний процес у міру свого розвитку зазнає змін.

У реальних процесів з плином часу можуть помітно змінюватися математичне очікування, дисперсія, закон розподілу, але процеси, у яких ці характеристики змінюються дуже повільно, з тією чи іншою ймовірністю відносяться до стаціонарних процесів. В даному випадку поняття "дуже повільно" означає, що зміни характеристик процесу на кінцевому інтервалі спостереження виявляються настільки незначними, що цими змінами можна знехтувати.

Також, варто зазначити, що чим коротшим є доступний інтервал спостереження (коротка вибірка часової послідовності), тим вищою є ймовірність прийняття хибного рішення про стаціонарність процесу. З іншого боку, якщо потрібно виконати прогноз на короткий інтервал, скорочення розміру вибірки в деяких випадках може привести до збільшення точності такого прогнозу [148].

Якщо процес зазнає змін, то параметри послідовності, визначені на інтервалі спостереження, за його межами будуть змінюватися. Таким чином, чим довший інтервал прогнозу, тим більший вплив на похибку прогнозування буде надавати мінливість характеристик послідовності. Цей факт змушує обмежитися лише короткотерміновим прогнозом, сильне скорочення інтервалу прогнозу дозволяє очікувати, що характеристики послідовності, які повільно змінюються не внесуть в прогноз істотних похибок.

У дисертаційній роботі прогнозування часової послідовності ПЗб АП виконується на даних моніторингу повітряного середовища, отриманих із центральної геофізичної обсерваторії імені Б. Срезневського, що знаходиться у у місті Києві [149]. Виміряні на одному з пунктів моніторингу забруднення повітряного середовища концентрації ПЗб атмосферного повітря, окрім

вуглекислого газу, зображено на рисунку 3.2. Концентрації вуглекислого газу, як і деяких інших параметрів, зображено окремо для детальнішої візуалізації наявності чи відсутності стаціонарності на рисунках 3.3а. – 3.3в.

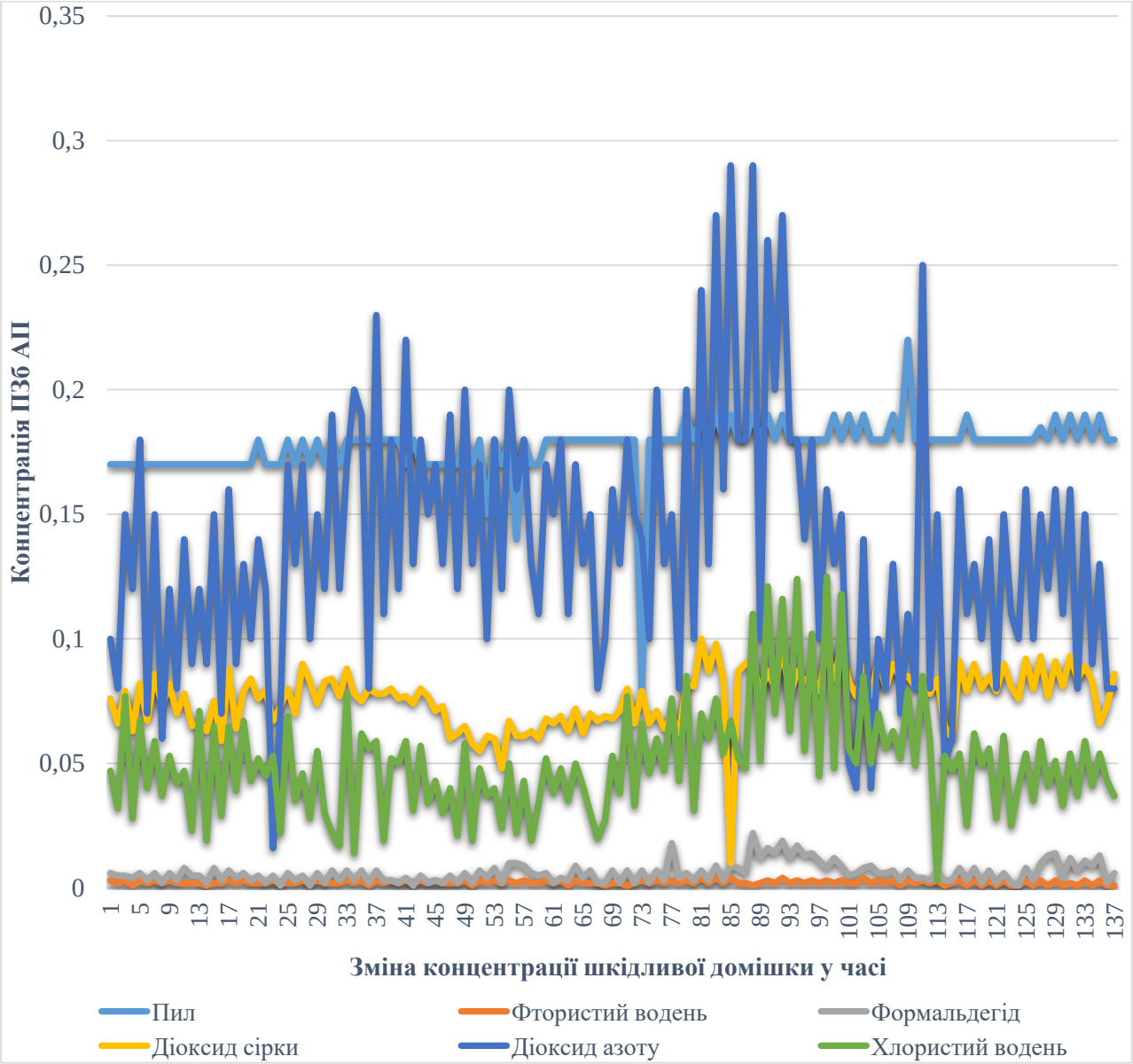


Рис. 3.2. Забруднення атмосферного повітря в м. Київ напротязі двох місяців

Отже, з рисунку 3.2. видно, що концентрації зображених ПЗб атмосферного повітря коливаються в межах десятків від цілого числа та часто повторюються, тому більшіть показників шкідливих домішок є стаціонарними. Детальніше стаціонарність можна розгледіти на рисунках 3.3а. – 3.3в.

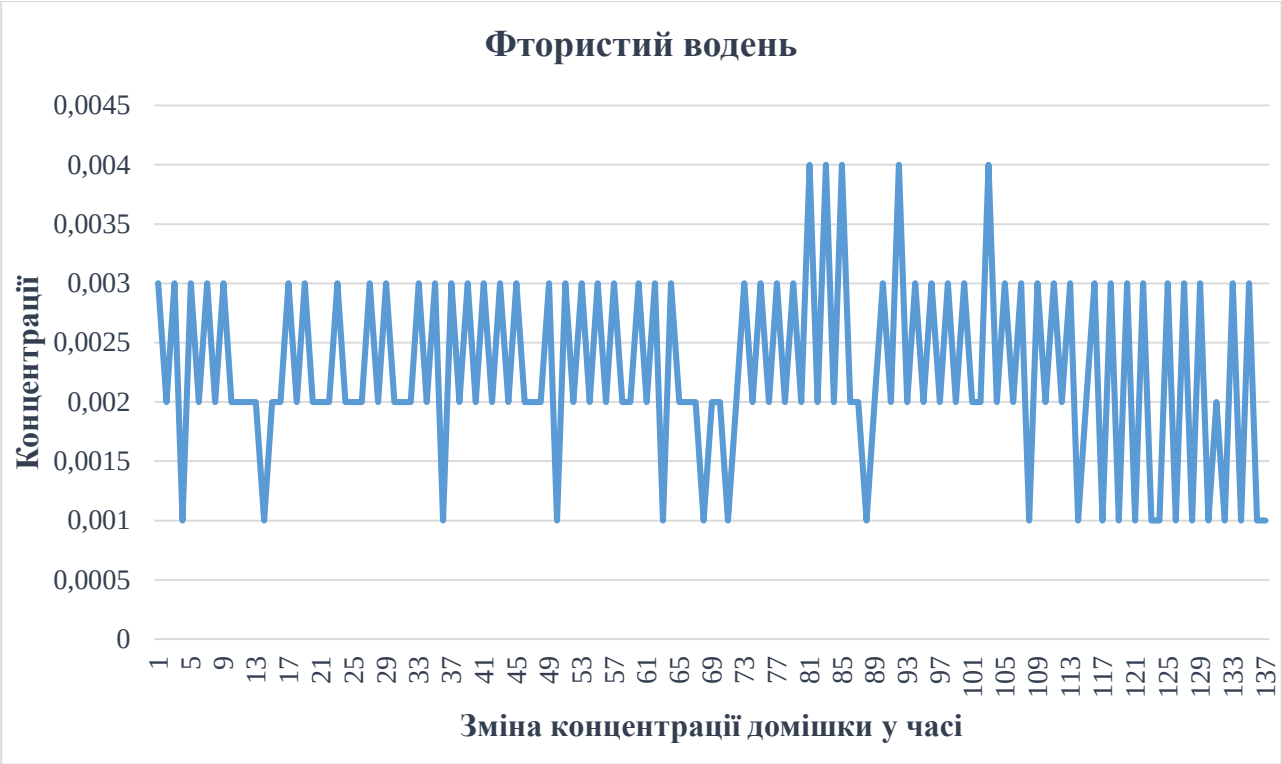


Рис. 3.3а. Концентрації викидів фтористого водню в часі

Рис. 3.3б. Концентрації викидів формальдегіду в часі

З рисунку 3.3а. зі стовідсотковою точністю можна зробити висновок, що часова послідовність фтористого водню є стаціонарною, оскільки відразу візуально помітним є постійне повторення однакових показників концентрації

фтористого водню у атмосферному повітрі. На рисунку 3.3б. зображено часову послідовність формальдегіду, де концентрації часто повторюються, але разом з тим різниця у концентраціях не становить більше чверті від одиниці, що також є показником стаціонарності.

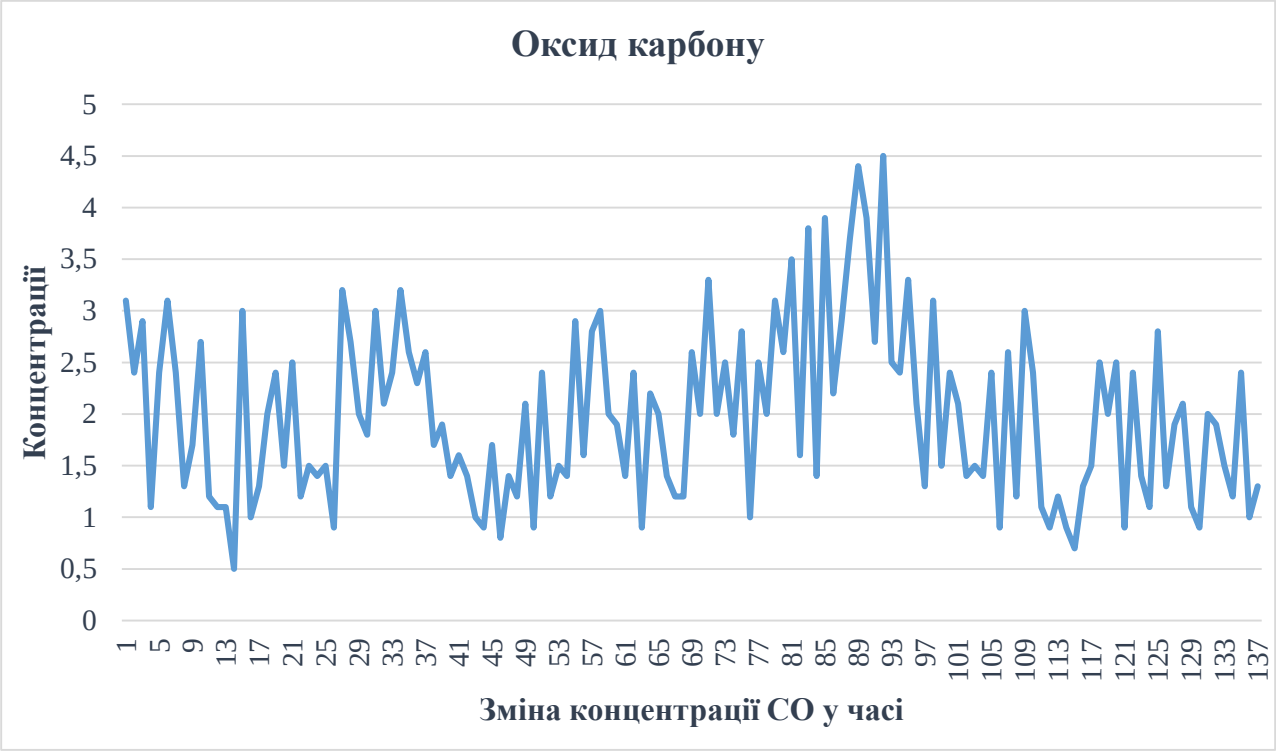


Рис. 3.3в. Концентрації викидів оксиду карбону в часі

Хоча концентрації оксиду карбону міняються в межах від 0,5 до 4,5, що видно на рисунку 3.3в., часова послідовність все одно вважається стаціонарною через невеликий проміжок.

Отже, всі часові послідовності параметрів забруднення АП зі стаціонарного посту моніторингу повітряного середовища є стаціонарними. Це призводить до того, що при оцінці по інтервалу спостереження отримується деяке усереднене їх значення, яке на цьому інтервалі залишалишається майже незмінними. Повністю усунути це явище, не можливо, але його можна послабити, якщо використати метод часових вікон для навчання та застосування нейронних мереж чи нейроподібних структур. Таким чином в роботі використаємо метод ковзних часових вікон для створення навчальних та тестових вибірок концентрацій для кожного ПЗб АП окремо.

3.1.2. Визначення наявності періодичності та тренду забруднення повітряного середовища шкідливими домішками

Метою аналізу тренду є розкладання часового ряду на головні компоненти, вимірювання розвитку кожної компоненти в минулому і її екстраполяція в майбутньому. Трендом називають загальну тенденцію при різнонаправленому русі, визначену спрямованістю зміни показників часового ряду [150].

Найпростішим прикладом тривалої зміни випадкового процесу є сума трьох компонент, відповідно до якого спостереження в момент t, це випадкова змінна X_t, що може бути визначена за формулою (3.7):

$$X_t = f(t) + g(t) + h\ , \tag{3.7}$$

де $f(t)$ - детермінована компонента (аналітична функція, яка має тенденцію до ряду динаміки); $g(t)$ - стохастична компонента (моделює характер періодичної варіації досліджуваного явища); h - випадкова компонента шуму.

Таким чином, виконавши віднімання тренду з досліджуваної часової послідовності динаміки є зміною масштабу даних і зберігає повну інформацію про варіації явища.

Для виділення тренду часових послідовностей використовують різні методи, такі як, згладжування ковзними середніми, частотну фільтрацію і т.п. [150]. Однак ці методи придатні лише для усереднення значень послідовності по точках деякої околиці і не можуть бути використані для прогнозування (екстраполяції) динамічних рядів, оскільки не дають в явному вигляді розрахункового рівняння детермінованої компоненти $f(t)$. Тому для виконання короткострокового прогнозування параметрів забруднення атмосферного повітря вибрано нейроподібні структури моделі послідовних геометричних перетворень, де виділення тренду закладено в алгоритм. Під час налаштування параметрів НС МПГП в режимі навчання використовується одна головна компонента, що дозволяє виконати виділення тренду в режимі застосування.

3.1.3. Вибір способу прогнозування параметрів забруднення атмосферного повітря

Практичне застосування того чи іншого методу прогнозування визначається такими факторами, як об'єкт прогнозу, складність і структура системи, наявність вихідної інформації і кваліфікація прогнозиста. Динаміку досліджуваних ПЗб АП можна прогнозувати за допомогою двох груп методів: однопараметричного і багатопараметричного прогнозування [151]. Загальним для обох груп є перш за все те, що застосовувані для одно- чи багато-параметричного прогнозування математичні функції ґрунтуються на оцінці вимірюваних значень минулого періоду (ретроспективи).

Однопараметричне прогнозування базується на функціональній залежності між прогнозованим параметром (змінної) і його минулим значенням, або фактором часу, тобто має місце залежність (3.8):

$$Y_{i+1} = f(t, Y_i, Y_{i-1}, \dots, Y_{i-n}) \tag{3.8}$$

де Y_{i+1} – прогнозоване значення; t – час;

Y_i – фактичне значення за останній період;

Y_{i-1} – фактичне значення за попередній період;

n – кількість періодів.

При побудові та обробці таких прогнозів у роботі пропонується використовувати методи трендового аналізу і експоненціального згладжування.

Аналіз багатопараметричних прогнозів показав, що в його основі лежить припущення про причинний взаємозв'язок між прогнозованим параметром і декількома іншими незалежними змінними (3.9):

$$Y_{i+1} = f(x_1, x_2, \dots, x_n) \tag{3.9}$$

де $x_1, x_2, \dots, x_n$ – параметри досліджуваного процесу.

Слід зазначити, що використання багатопараметричних методів на практиці виправдано для середньо- і довготермінового прогнозування, коли на зміну прогнозованої змінної впливає зміна багатьох незалежних змінних. Для

короткотермінового (вибраного у роботі) прогнозування показників, що змінюються щодня, як правило, користуються однопараметричними методами. Тому, в роботі реалізовується саме метод однопараметричного прогнозування ПЗб атмосферного повітря.

Архітектура нейроподібної структури МПГП для виконання завдання однопараметричного прогнозування складається із *n* входів та одного виходу, що зображено на рисунку 3.4.

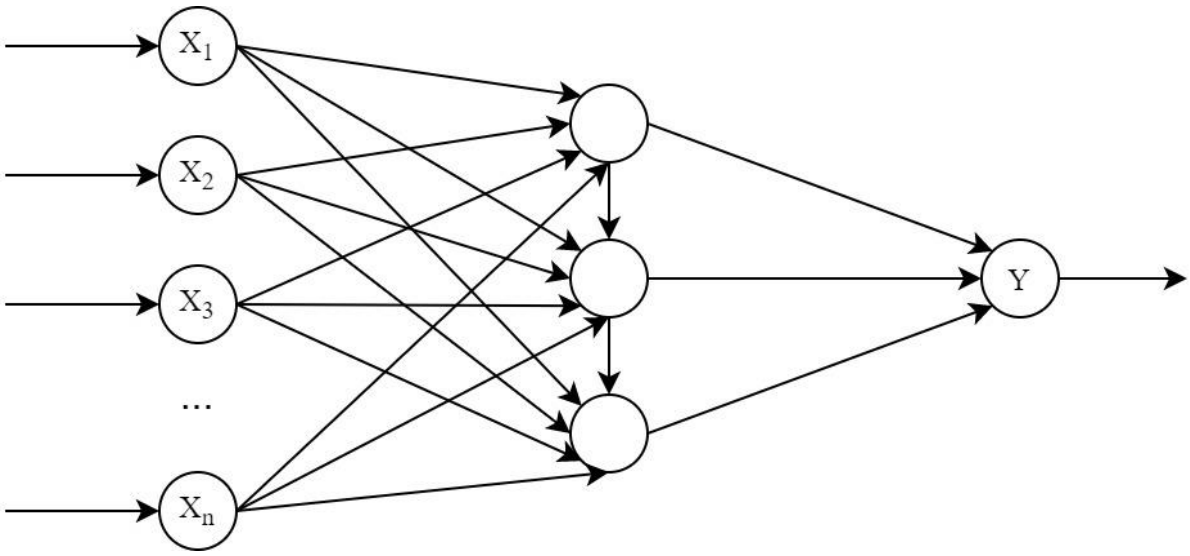


Рис. 3.4. Схема нейроподібної структури МПГП для виконання однопараметричного прогнозування

Оскільки в попередньому параграфі визначено, що концентрації ПЗб АП є стаціонарними, тому навчальні та тестові вибірки для виконання завдання прогнозування формуються за принципом “ковзних часових вікон”, використовуючи два часових вікна, що відповідають за вхідний та вихідний вектори даних. Часові вікна переміщаються з кроком ковзання *k* по даних часової послідовності, завдяки чому пара вхідного та вихідного векторів приймається як реалізація навчальної пари [152]. Ширина вікон та крок зміщення визначаютсья шляхом проведення експериментів. Якщо вихідне вікно має розмірність *p+1* і крок ковзання *k = 1*, тоді сформована множина реалізацій виглядає так, як показано в таблиці 3.1.

Таблиця 3.1.

Навчальна пара	Входи				Виходи
1	$X(t_1)$	$X(t_2)$	...	$X(t_p)$	$X(t_{p+1})$
2	$X(t_2)$	$X(t_3)$	...	$X(t_{p+1})$	$X(t_{p+2})$
...	...	...	...	...	...
n	$X(t_n)$	$X(t_{n+1})$	...	$X(t_{p+n-1})$	$X(t_{p+n})$

Однією з важливих процедур вирішення завдання нейромережевого прогнозування ПЗб АП є вибір способу прогнозування. У роботі виконано два способи прогнозування часових послідовностей за допомогою комітету нейроподібних структур різних типів [153]: однокрокове та багатокрокове прогнозування.

Проаналізовано, що однокрокове прогнозування використовується для оперативних та для певних випадків короткотермінових прогнозів, звичайно абсолютних значень послідовності. Прогнозування здійснюється тільки на один крок вперед, але для прогнозування на наступному кроці використовується реальне, а не прогнозоване значення. Результатом однокрокового прогнозування є відображення вхідного вектора концентрацій у вихідному (реальний вхід -> прогнозований вихід). Схема однокрокового прогнозування на основі використання НС зображена на рисунку 3.5.

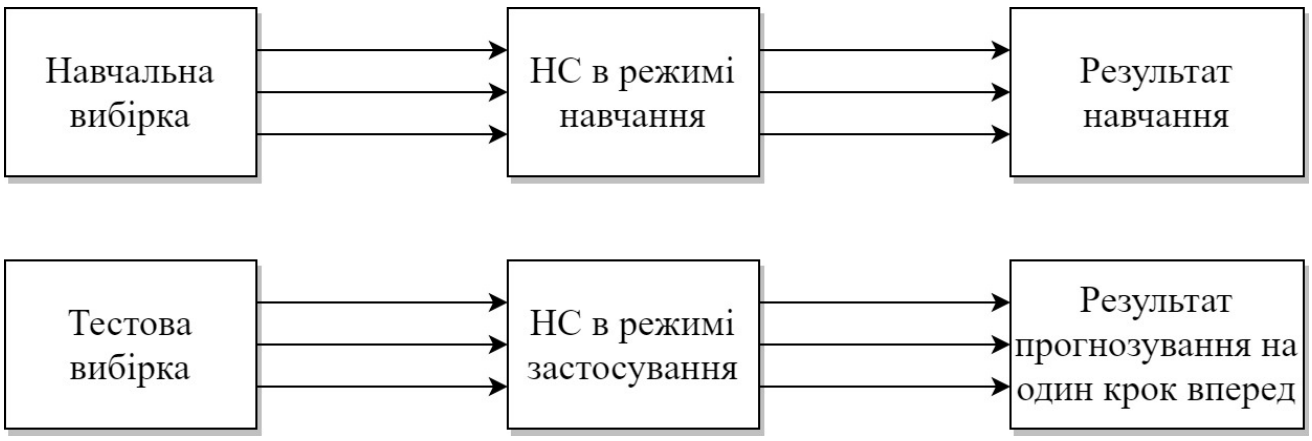


Рис. 3.5. Схема виконання однокрокового прогнозу за допомогою нейроподібних структур

Після здійснення однокровокого прогнозування невідомого значення $X(t_{p+n})$ (табл 3.1) стоїть завдання виконати багатокрококе прогнозування, що застосовується для коротко-, середньо-, та довготермінових прогнозів. Багатокрокове прогнозування призначене для визначення тренду і головних компонент (точок) зміни тренду для певного проміжку часу в майбутньому. При цьому НС використовує отримані вихідні дані для моментів часу $X(t_{p+n+1})$, $X(t_{p+n+2})$, ... , $X(t_{p+n+m})$ в якості вхідних даних для прогнозування у моменти часу $X(t_{p+n+2})$, ... , $X(t_{p+n+m+1})$. Загальна схема багатокрокового прогнозування за допомогою НС зображена на рисунку 3.6. [154].

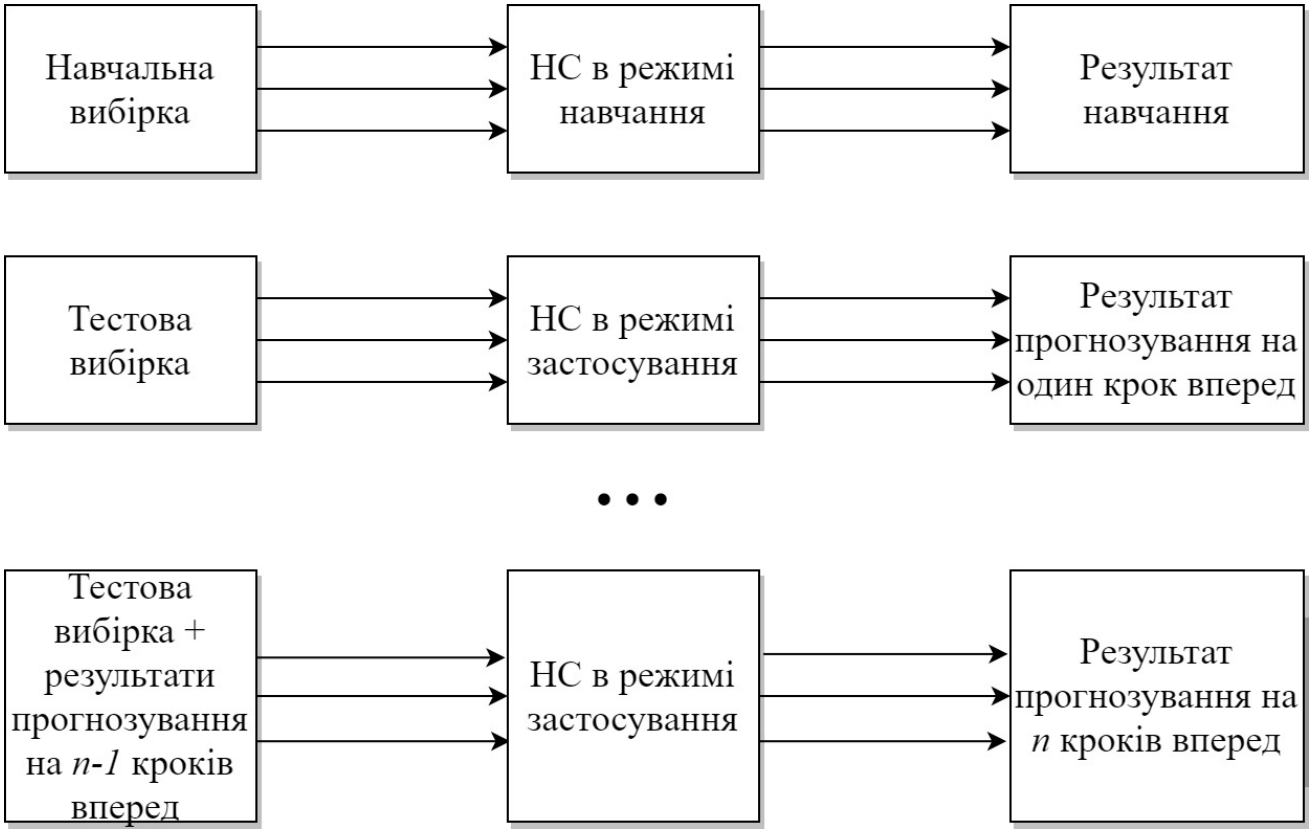


Рис. 3.6. Схема виконання багатокрокового прогнозу за допомогою нейроподібних структур

Отже, з рис. 3.6. видно, що першим етапом виконання короткотермінового прогнозу є однокрокове прогнозування, а наступним етапом є виконання багатокрокового прогноузвання на n кроків вперед. Таким чином необідно описати обидва етапи реалізації короткотермінового прогнозування тренду забруднення атмосферного повітря на основі вибаної моделі НС МПГП.

3.1.4. Однокрокове прогнозування параметрів забруднення повітряного середовища на основі НС МПГП

Здібності нейронної мережі до прогнозування безпосередньо випливають з її здатності до узагальнення і виділення прихованих залежностей між вхідними та вихідними даними. Після навчання мережа здатна прогнозувати майбутнє значення певної послідовності на основі декількох попередніх значень і (або) якихось існуючих зараз чинників. Слід зазначити, що прогнозування можливо тільки тоді, коли попередні зміни дійсно в певному ступені визначають майбутнє.

У параграфі розділу розглядаються методи однокрокового та багатокрокового прогнозування часової послідовності на основі НС МПГП, котра забезпечує автоматичне розкладання часових рядів на дві складові: тренд (тенденцію зміни) та коливання різної частоти. Коливання складаються з суми всіх виділених складових, що рівна відлікам часової послідовності.

Власне, метод однокрокового прогнозування ПЗб атмосферного повітря за допомогою НС МПГП включає в себе наступні етапи:

Етап 1. Згладження концентрацій деякого ПЗб АП на основі виділення тренду за допомогою навчання НС МПГП з однією головною компонентою.

Етап 2. Формування матриць значень для концентрацій певного ПЗб із деякого посту моніторингу викидів шкідливих домішок у атмосферне повітря, використовуючи різну кількості часових вікон за методом ковзного середнього (від 5 до 30 з кроком 5).

Етап 3. Розподіл сформованих матриць на навчальні і тестові вибірки даних.

Етап 4. Застосування нейроподібної структури МПГП на всіх матрицях з різною кількістю часових вікон для виконання однокрокового прогнозування забруднення повітря певним параметром забруднення повітря.

Етап 5. Визначення методом перебору, числа часових вікон, при якому результати прогнозування є найточнішими за допомогою розрахунку похибок прогнозування.

Зупинимося детальніше на етапі згладження концентрацій деякого параметру забруднення атмосферного повітря.

3.1.4.1. Згладження вихідних значень навчальної вибірки даних

Згладження виходів навчальної вибірки концентрацій параметрів забруднення атомсферного повітря виконується набором наступної послідовності кроків:

Крок 1. Виконання навчання НС1 моделі послідовних геометричних перетворень лінійного типу з використанням векторів навчальної вибірки концентрацій параметрів забруднення повітряного середовища виду $x_{i1}, \dots, x_{ij}, \dots, x_{in}, Y_i^{Real} \rightarrow Y_i^{Real}$, де Y_i^{Real} – відомий вихід навчального вектора концентрацій.

Крок 2. У випадку, якщо число нейронних елементів (НЕ) прихованого шару є рівним кількості входів вектора n, в режимі застосування на виході НС1 моделі послідовних геометричних перетворень отримується точне значення сигналу Y_i^{Real}.

Крок 3. Зменшуючи число НЕ прихованого шару НС1 моделі послідовних геометричних перетворень на одиницю, отримується так зване згладжене значення сигналу Y_i^{sReal}, яке не враховує останньої головної компоненти при перетвореннях з початкової системи координат в систему головних компонент НС1.

Крок 4. З отриманого згладженого сигналу Y_i^{sReal} виконується формування нової матриці векторів концентрацій певного параметру забруднення атмосферного повітря виду $x_{i1}, \dots, x_{ij}, \dots, x_{in}, \rightarrow Y_i^{sReal}$, де Y_i^{sReal} – згладжений вихід навчального вектора концентрацій.

Формування нової вибірки даних моніторингу повітряного середовища відбувається за допомогою використання методу ковзних часових вікон з часовим вікном n – число часових вікон, що визначається методом перебору.

Наступним етапом є застосування сформованих вибірок концентрацій параметрів забруднення атмосферного повітря зі згладженими вихідними значеннями та коректування відхилень для зменшення похибок прогнозування та розширення горизонту прогнозування.

3.1.4.2. Корекція похибок комітетом НС різних типів для розширення горизонту прогнозування параметрів забруднення атмосферного повітря

Кожен метод прогнозування даних результує з деякими відхиленнями, що називаються похибками прогнозування. У випадку коректного застосування засобів машинного навчання і адекватних тренувальних даних систематичні складові відхилень від точних значень вихідних величин виключаються.

Загальна похибка функціонування оцінюється зваженою сумою квадратів, або абсолютних величин відхилень різних знаків. Більшість методів навчання, починаючи від методів ідентифікації коефіцієнтів поліномів і до штучних нейронних мереж використовують принципи підбору параметрів на основі мінімізації середньо-квадратичних відхилень (методу найменших квадратів).

У дисертаційній роботі розроблено та досліджено метод прогнозування параметрів забруднення атмосферного повітря, котрий також базується на зменшенні відхилень за рахунок виконання корекції знайдених похибок навчання та застосування нейроподібних структур моделі послідовних геометричних перетворень. Розглянемо розроблений метод короткотермінового прогнозування параметрів забруднення повітряного середовища на основі коректування похибок апроксимації нелінійних поверхонь відгуку.

Розроблений метод короткотермінового прогнозування побудований на врахуванні передумови, що відхилення від точних значень для нелінійної апроксимації в більшості точок відгуку є меншими ніж для лінійної. Коригування похибки передбачення для збільшення точності прогнозування реалізується на застосуванні комітету нейроподібних структур моделі послідовних геометричних перетворень різних типів, кожна з яких послідовно виконує покладені на неї функції.

Користуючись відомими значеннями y_i (що можливе лише для векторів тренувальної вибірки), для довільної точки виконується співвідношення за формулою (3.10):

$$y_i = y_i^{RBF} - \alpha(y_i^{RBF} - y_i^{SGTM}) \tag{3.10}$$

де y_i^{SGTM} - значення сигналу, отримане на виході навченої НС МПГП лінійного типу; α - коефіцієнт пропорційності, який підбирається експериментально і є однаковим для всіх точок вибірки.

Ілюстрацію принципу такого перетворення для випадку одновимірного сигналу подано на рис.3.7.

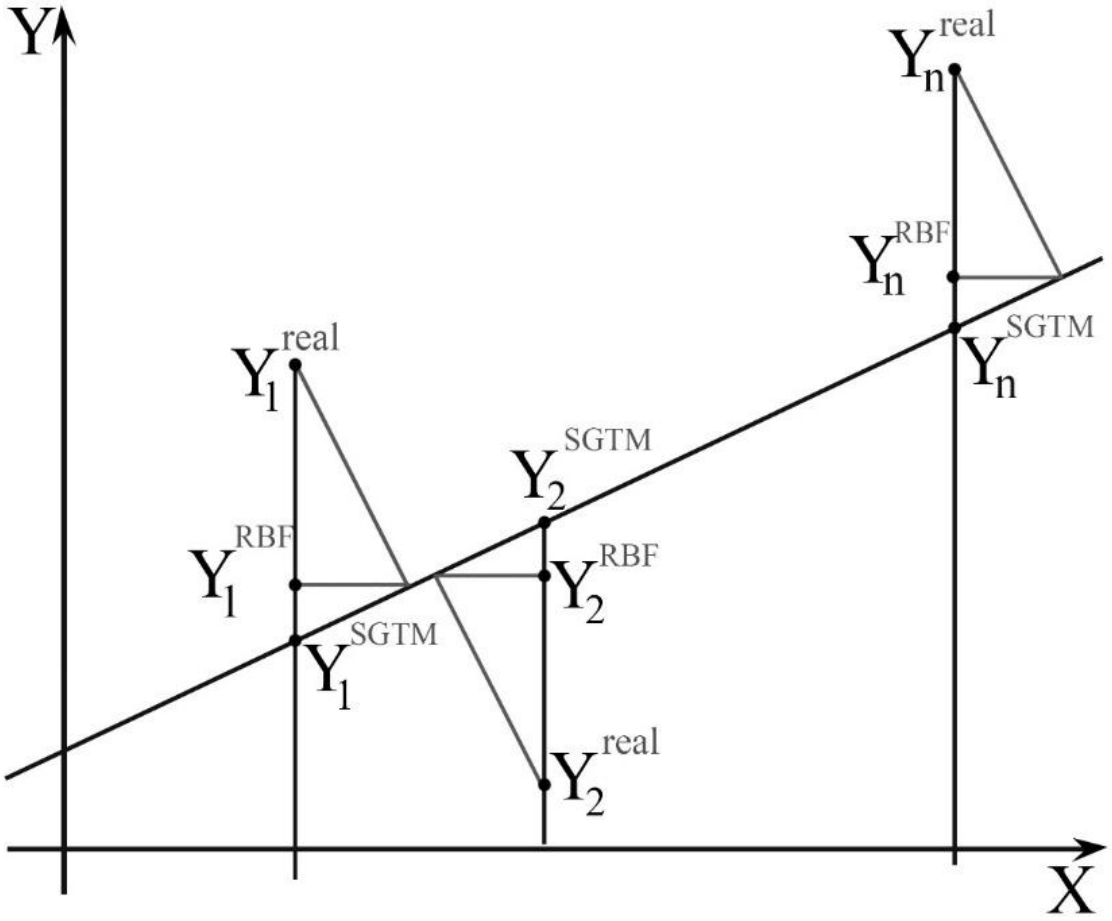


Рис. 3.7. Графічне зображення обрахунку відхилень розробленим методом корекції похибки у випадку одновимірного сигналу

Величина сигналу y_i^{RBF} може бути передбачена за допомогою НС МПГП нелінійного типу, зокрема з використанням поліноміальних, або RBF входів розширення. Так як передбачаємо згладжене значення виходу y_i^{RBF}, в якому в навчальній вибірці усувається остання, як правило шумова складова, точність передбачення вдається помітно підвищити. В результаті виникає можливість окремо зменшувати похибки різних знаків і помітно підвищити точність прогнозів.

3.1.5. Багатокрокове прогнозування параметрів забруднення повітряного середовища

Одним із завдань дисертаційної роботи є розробка короткотермінового прогнозування концентрацій параметрів забруднення атмосферного повітря. Для цього виконується дослідження методів однокрокового прогнозування, що описано в попередніх параграфах розділу, та методів багатокрокового прогнозування.

Багатокровоке прогнозування параметрів забруднення АП виконується допоки не буде досягнено горизонту прогнозування, що є параметром моделі прогнозування на основі методу ковзних часових вікон. Горизонт прогнозування визначає розміри часових інтервалів (днів, тижів, місяців і т.д.) для побудови деякого прогнозу. Оскільки завданням є розробка методу короткотермінового прогнозування, а досліджувані концентрації ПЗб атмосферного повітря виміряні двічі на день, тому ймовірним горизонтом прогнозування є два тижні з інтервалом прогнозування двічі на день.

Також, горизонт прогнозування є часовим інтервалом, в межах якого реалізується прогнозування ПЗб АП із заданою точністю. Тому у дослідженні горизонт прогнозування визначаєтся за домопогою знаходження останнього показника концентрації деякого параметру забруднення повітряного середовища, похибка багатокрокового прогнозування якого наближається до 10 %.

Розробка та реалізація методу коректування похибки прогнозування на основі комітету нейроподібних структур різних типів збільшує точність однокрокового прогнозування. Багатокрокове прогнозування параметрів забруднення АП виконується за рахунок перенавчання НС МПГП з використанням результатів однокрокового прогнозування. Тому відбувається збільшення також точності багатокрокового прогнозування, яке виконується поки не досягнуто визначеного горизонту прогнозування.

Таким чином, покращення точності однокрокового прогнозування, підвищує точність багатокрового прогнозування, чим розширює горизонт прогнозування.

3.2. Метод нейромережевої ідентифікації коефіцієнтів полінома для прогнозування параметрів забруднення атмосферного повітря

Для моделювання поверхонь відгуків в задачах прогнозування параметрів забруднення атмосферного повітря, найчастіше використовують багатошаровий перцептрон, радіально-базисні мережі, нейронні мережі узагальненої регресії, машину опорних векторів [87 - 113]. Також використовують нейроподібну структуру моделі послідовних геометричних перетворень різних типів.

У параграфі 3.1. розроблено та описано метод короткотермінового прогнозування параметрів забруднення атмосферного повітря за допомогою комітету лінійних і нелінійних нейроподібних структур для часткового коректування окремо додатних і від'ємних відхилень від точних значень, що забезпечує збільшення горизонту прогнозування часових послідовностей. У параграфі 3.2., на противагу існуючим методам прогнозування параметрів забруднення АП, пропонується метод прогнозування на основі нейромережевої ідентифікації коефіцієнтів поліномів із застосуванням нейроподібної структури моделі послідовних геометричних перетворень лінійного типу. Згідно з ним, спочатку здійснюється навчання обраної НС, а в режимі застосування обчислюються лінійні поліноми, з рахунок яких відбувається прогнозування параметрів забруднення амосферного повітря.

Таким чином реалізація нейромережевої ідентифікації коефіцієнтів поліномів за допомогою НС МПГП для задачі прогнозування параметрів забруднення повітряного середовища включає в себе використання режимів навчання та застосування. Хоча навчання НС МПГП виконується у повністю автоматичному режимі, а її використання у режимі застосування є ще швидшим [155], але цього не достатньо для прогнозування параметрів забруднення атмосферного повітря на мобільних пристроях та мікроконтролерах.

Використання комітету НС різних типів збільшує обчислювальні ресурси чим збільшує затрати опертивної пам'яті та часові затримки. Також використання нейромережевих та нейроподібних методів прогнозування вимагає наяності значної частини місця зберігання інформації на необхідному пристрої. Оскільки

мова йде про використання мобільних пристроїв та автомномних пристроїв на основі мікроконтролерів, тому виникає необхідність зменшення затрат оперативної пам’яті для збільшення швидкості прогнозування параметрів забруднення атмосферного повітря.

3.2.1. Метод прогнозування параметрів забруднення атмосферного повітря шляхом нейромережевої ідентифікації коефіцієнтів поліномів

Для зменшення затрат оперативної пам’яті під час прогнозування параметрів забруднення атмосферного повітря на мобільних пристроях можна використати поєднання нейроподібних структур та поліномів. Для розрахунку поліномів використовуються коефіцієнти, які отримуються за допомогою навчання нейроподібної структури моделі послідовних геометричних перетворень з багатьма виходами. Ідентиціковані коефіцієнти вносяться в матричний оператор. Ця процедура розроблена І. В. Ізоніним для отримання коефіцієнтів ваг синаптичних зв’язків з НС МПГП з багатьма виходами [156] для використання роздільної здатності зображень. Однак нейромережеву ідентифікацію коефіцієнтів поліномів можна використати також у задачах прогнозування.

Оскільки, під час виконання нейромережевої ідентифікації коефіцієнтів поліномів для прогнозування ПЗб повітряного середовища використовується НП МПГП, необхідним є опис її побудови. Отже побудова НС МПГП складається з наступних кроків [157]:

Крок 1. Вибір початкової конфігурації мережі (наприклад, один проміжний шар з числом елементів в ньому).

Крок 2. Проведення ряду експериментів з різними конфігураціями, запам'ятовуючи при цьому кращу мережу (в сенсі контрольної помилки). Для кожної конфігурації слід провести кілька експериментів, щоб не отримати помилковий результат через те, що процес навчання потрапив в локальний мінімум.

Крок 3. Якщо в черговому експерименті спостерігається так звана "недонавченість" (мережа не видає результат прийнятної якості), потрібно спробувати додати додаткові нейрони в проміжний шар чи шари, або додати новий проміжний шар.

Нейроподібна структура моделі послідовних геометричних перетворень з багатьма виходами зображена на рисунку 3.8.

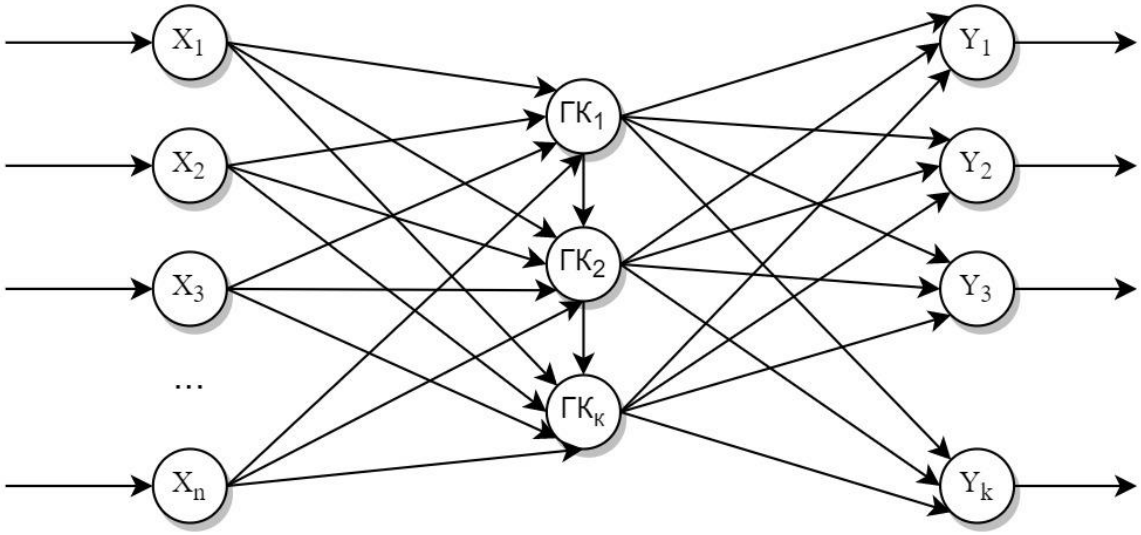


Рис.3.8. НС МПГП з багатьма виходами

Після побудови НС МПГП (рис. 3.8.) потрібно реалізувати алгоритм нейромережевої ідентифікації коефіцієнтів полінома [156]. Першим кроком виконується навчання НС МПГП на вибірці векторів концентрацій параметрів забруднення повітряного середовища. Під час навчання нейроподібної структури моделі послідовних геометричних перетворень формується матриця коефіцієнтів ($V^{(m)}$) площини відгуку $\alpha_{i,j}$ [157] за формулою (3.11):

$$V^{(m)} = [\alpha_{i,j}]_{i=\overline{1,k^2+1}}^{j=\overline{1,(km)^2}} \tag{3.11}$$

Наступним кроком виконується формування матриця коефіцієнтів, котрі в подальшому можуть бути використані для створення матриці коефіцієнтів синаптичних ваг звязків з НС МПГП з багатьма виходами. Проте у дослідженні сформована матриця коефіцієнтів використовується для розрахунку лінійного полінома.

3.2.2. Розвинення методу прогнозування параметрів забруднення повітряного середовища

Для зменшення часових та ресурсних(обчислювальних) затрат під час прогнозування параметрів забруднення атмосферного повітря при надходженні даних в реальному часі на мобільних пристроях та мікроконтролерах отримав подальший розиток метод ідентифікації коефіцієнтів ваг синаптичних зв'язків нейроподібної структури моделі послідовних геометричних перетворень з кількома виходами.

Існуючий метод розвинено шляхом побудови матриці коефіцієнтів апроксимаційних поліномів, створеної шляхом їх ідентифікації за результатами навчання лінійних нейроподібних структур моделі послідовних геометричних перетворень з одним виходом. Використання НС МПГП з одним виходом замість декількох виходів забезпечує пришвидшення виконання прогнозування ПЗб АП за рахунок зменшення затрат обчислювальних ресурсів, а тому зменшення затрат оперативної пам'яті та збільшення швидкості прогнозуваня.

Серед низки різних видів поліномів було досліджено полноми Вінера, поліноми Лагранжа, поліноми Колмогорова-Габора, лінійні поліноми та інші. В дисертаційній роботі для подальшого розвитку нейромережевої ідентифікації коефіцієнтів поліномів вибрано лінійний поліном замість, оскільки його використання не призводить до збільшення затрат оперативної пам'яті.

Після навчання НС МПГП лінійного типу на матриці навчальних даних, виконується її застосування на тестовій матриці *T*, - діагональній матриці розмірність якої рівна розмірності вхідного вектора для навчання та одного додаткового вектора, елементами якого є нулі. Слід зазначити, що останній є першим вектором тестової матриці, яку наведено в таблиці 3.2.

Метою такого кроку є отримання коефіцієнтів лінійного полінома, який буде використовуватися на стадії застосування методу прогнозування параметрів забруднення атмосферного повітря без використання нейроподібної структури моделі послідовних геометричних перетворень.

Таблиця 3.2.

Діагональна матриця

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	…	X_i
0	0	0	0	0	0	0	0	0	...	0
1	0	0	0	0	0	0	0	0	...	0
0	1	0	0	0	0	0	0	0	...	0
0	0	1	0	0	0	0	0	0	...	0
0	0	0	1	0	0	0	0	0	...	0
0	0	0	0	1	0	0	0	0	...	0
0	0	0	0	0	1	0	0	0	...	0
0	0	0	0	0	0	1	0	0	...	0
0	0	0	0	0	0	0	1	0	...	0
0	0	0	0	0	0	0	0	1	...	0
...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	...	1

Результатом застосування нейроподібної структури моделі послідовних геометричних перетворень на тестовій матриці, є отримання коефіцієнтів полінома для використання під час виконання наступного етапу поставленої задачі – прогнозування параметрів забруднення атмосферного повітря. Передбачені виходи використовуємо як коефіцієнти a_i для лінійного полінома (3.12) під час розвязання поставленої задачі:

$$y_i = a_0 + a_1 * x_1 + \cdots + a_i * x_i \qquad (3.12)$$

де коефіцієнти $a_0 = y_0$, $a_i = y_i - y_0$, а x_i – вхідні параметри тестової матриці, для якої потрібно виконати передбачення виходів, тобто відновити пропущені дані. Таким чином виконується заповнення пропущених параметрів атмосферного повітря за допомогою НС МПГП та на її основі лінійних поліномів.

3.3. Порівняльна оцінка ефективності методів прогнозування параметрів забруднення атмосферного повітря

3.3.1. Опис експериментальної вибірки даних для завдання підвищення точності прогнозування параметрів забруднення повітря

У дослідженні використано дані спостереження за забрудненням атмосферного повітря в м. Києві. Систематичні спостереження за вмістом шкідливих речовин в атмосферному повітрі проводяться на 16 стаціонарних постах (ПСЗ) з періодичністю відбору проб 6 днів на тиждень, 3-4 рази на добу. Визначається 20 забруднюючих домішок, але для загального доступу на офіційній сторінці [149] подається лише дев’ять домішок.

Для різних постів спостереження публікується різна кількість виміряних домішок. Кожного тижня відбувається зміна оприлюднених спостережень, тому під час дисертаційного дослідження виконувався моніторинг щотижневих вимірювань, та на протязі двох місяців відбувався збір даних. Інформація про різні домішки та про стан забруднення атмосферного повітря наводиться по чотирьох стаціонарних постах. У дослідженні використані дані, котрі виміряні на стаціонарному пості №7 – Бесарабська площа та наведені в таблиці 3.3.

Таблиця 3.3.

Уривок даних забруднення м. Києва (Бесарабська площа) за два місяці 2019 року з веб-ресурсу центральної геофізичної обсерваторії імені Б.Срезневського

Solid	Sulfur	Nitrogen	Fluoride	Chloride	Formaldehyde	Carbon
0,18	0,086	0,08	0,001	0,037	0,006	1,3
0,18	0,073	0,08	0,001	0,043	0,002	1
0,19	0,066	0,13	0,003	0,054	0,013	2,4
0,18	0,083	0,09	0,001	0,041	0,009	1,2
0,19	0,089	0,15	0,003	0,059	0,011	1,5
0,18	0,084	0,08	0,001	0,037	0,007	1,9
0,19	0,093	0,16	0,002	0,054	0,012	2
0,18	0,081	0,11	0,001	0,033	0,004	0,9

Продовження Табл. 3.3.

0,18	0,076	0,1	0,001	0,041	0,001	1,1
0,18	0,081	0,11	0,001	0,025	0,002	1,4
0,18	0,09	0,15	0,003	0,061	0,006	2,4
0,18	0,079	0,08	0,001	0,028	0,002	0,9
0,18	0,085	0,14	0,003	0,056	0,007	2,5
0,18	0,08	0,1	0,001	0,049	0,002	2
0,18	0,09	0,13	0,003	0,062	0,008	2,5
0,19	0,079	0,11	0,001	0,025	0,003	1,5
0,18	0,091	0,16	0,003	0,054	0,008	1,3
0,18	0,06	0,06	0,002	0,047	0,003	0,7
0,18	0,065	0,05	0,001	0,053	0,003	0,9
0,18	0,084	0,15	0,003	0,003	0,006	1,2
0,18	0,078	0,08	0,002	0,059	0,003	0,9
0,18	0,081	0,25	0,003	0,085	0,004	1,1

Концентрації параметрів забруднення атмосферного повітря, що наведені в таблиці 3, зображено у вигляді діаграми на рисунку 3.9.

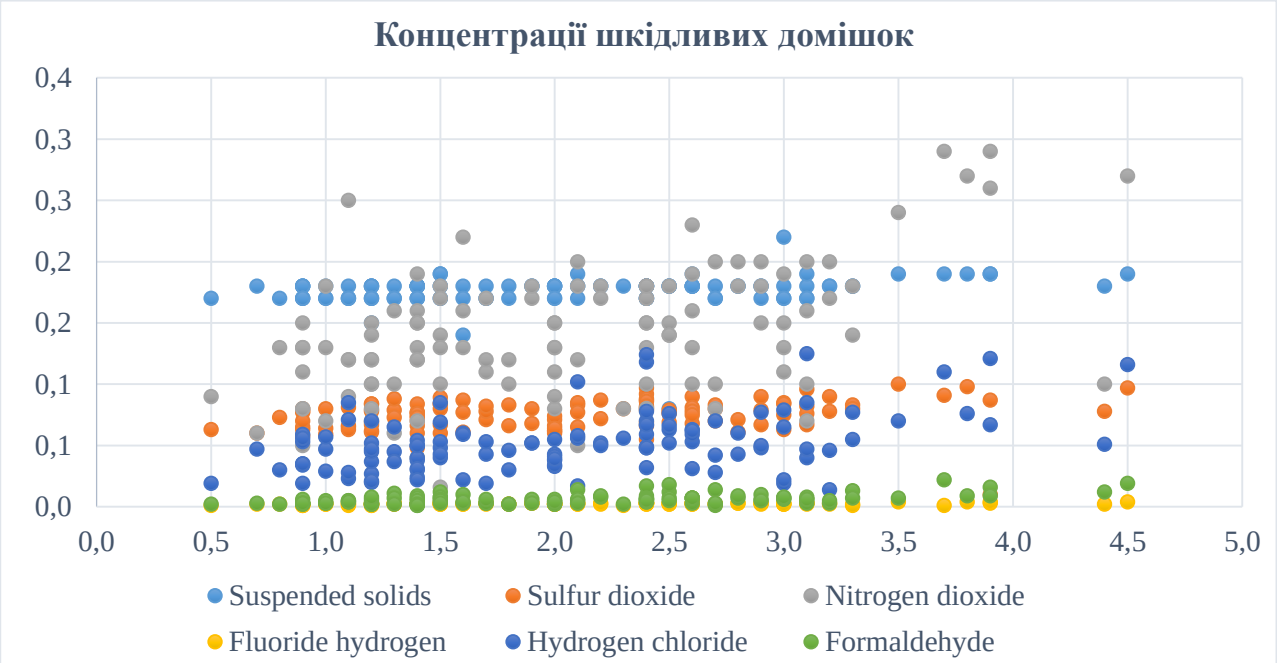


Рис. 3.9. Концентрації ПЗб АП в місті Києві за виміряний період часу

Зображені на рисунку 3.9. шкідливі викиди у атмосферне повітря є не стаціонарними, тому підлягають дослідженням.

3.3.2. Результати порівняння досліджуваних методів прогнозування параметрів забруднення повітряного середовища

Виконання задачі прогнозування різними складними методами може мати низькі відносну чи середньоквадратичну похибки, але все ж поступатися простому прогнозуванню наївним методом. Тому у роботі першим етапом виконано порівняння методу на основі НС МПГП з методом наївного прогнозу для підтвердження, що розроблюваний метод показує кращі результати прогнозування, а тому є сенс у його використанні.

Таким чином, у роботі виконано порівняння ефективності застосування наївного прогнозу та НС МПГП для прогнозування концентрацій у даних моніторингу атмосферного повітря. Прогнозування ПЗб АП реалізовано для таких параметрів забруднення як оксид карбону, діоксиди сірки та азоту, фтористий водень та формальдегід. Результати прогнозування концентрацій ПЗб АП за допомогою НС МПГП наведено у таблиці 3.4.

Таблиця 3.4.

Похибки прогнозування концентрацій ПЗб АП із застосуванням НС МПГП

	Параметри / Похибки	**MAPE, train**	**RMSE, train**	**RMSE_M, train**
НС МПГП	**Оксид вуглецю**	2,555780	0,007214	0,229040
	Діоксид азоту	2,637694	0,000515	1,110193
	Діоксид сірки	1,229528	0,000141	0,164614
	Фтористий водень	9,220656	0,000033	1,110193
	Формальдегід	6,502237	0,000050	0,423197
	Методи / Похибки	**MAPE, test**	**RMSE, test**	**RMSE_M, test**
	Оксид вуглецю	3,906991	0,017413	0,552818
	Діоксид азоту	3,411114	0,001201	0,571996
	Діоксид сірки	1,465118	0,000390	0,454599
	Фтористий водень	13,02806	0,000062	2,085392
	Формальдегід	6,904762	0,000081	0,681258

Оскільки розраховані відхилення за допомогою НС МПГП лінійного типу є найменшими саме для часового вікна $t_{iw} = 10$, тому саме цей інтервал використано для прогнозування конентрацій різних параметрів забруднення повітря.

Графічне представлення результатів застосування наївного прогнозу та НС МПГП для прогнозування параметрів забруднення повітряного середовища зображено на рисунку 3.10.

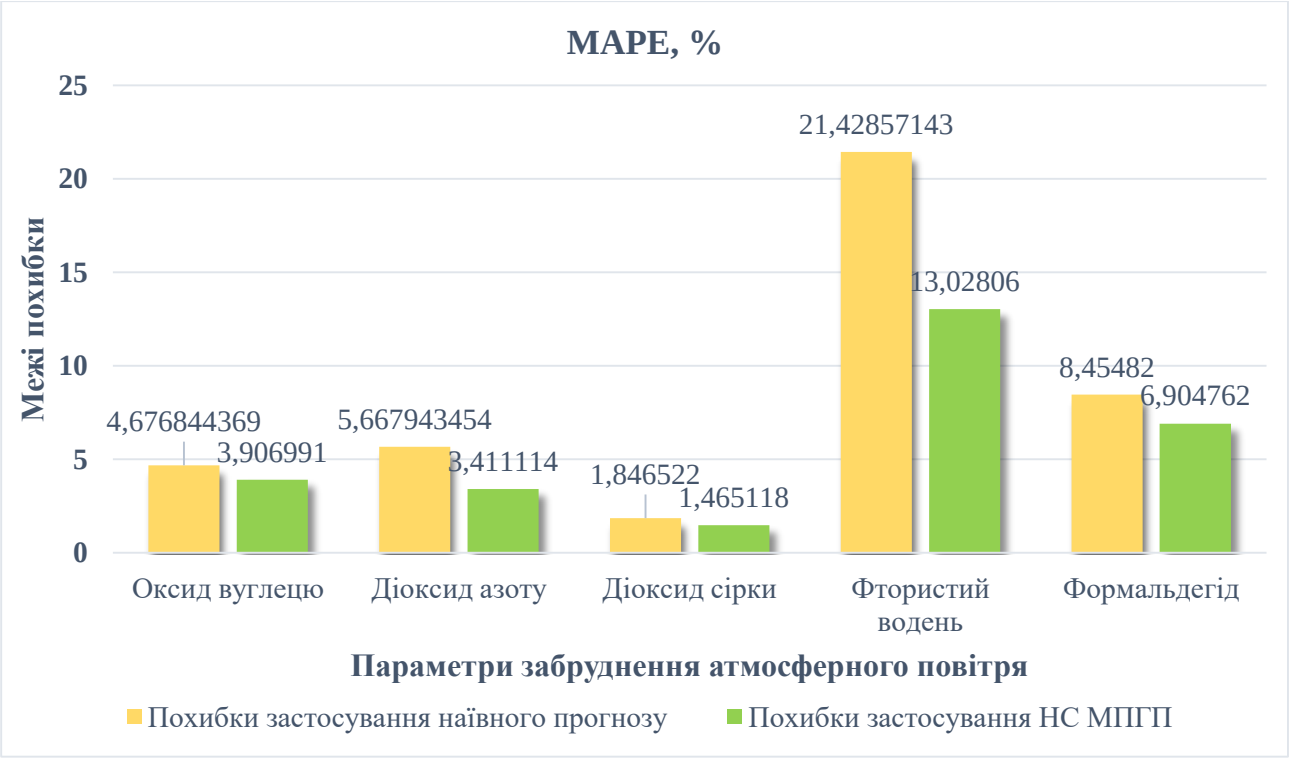


Рис. 3.10. Похибки MAPE під час застосування наївного прогнозу та методу прогнозування на основі НС МПГП

Отже, у дисертаційній роботі визначено, що метод прогнозування з використанням НС МПГП результує з нижчими похибками застосування, ніж наївний прогноз. Тому обгрунтованою є розробка та дослідження методу короткотермінового прогнозування з розширеним горизонтом прогнозування за допомогою комітету нейроподібних структур різних типів.

Для перевірки ефективності розроблених та удосконалених методів прогнозування параметрів забруднення повітряного середовища необхідно реалізувати регресійні та нейромережеві методи, вибір яких обгрунтовано в

першому розділі. Також потрібно виконати прогнозування параметрів забруднення атмосферного повітря на початкових експериментальних даних. Для того, щоб мати можливість визначити точність прогнозування параметрів забруднення АП виконуєтсья розділення векторів концентрацій ПЗб АП на навчальну і тестову вибірки даних, та виконання прогнозування різними методами.

3.3.2.1. Результати короткотермінового прогнозування концентацій оксиду карбону досліджуваними методами

Таким чином, у роботі виконано порівняння ефективності застосування наступних досліджуваних методів прогнозування концентрацій у даних моніторингу атмосферного повітря: методу на основі градієнтного стохастичного спуску, методу опорних векторів, дерева рішень, випадкового лісу, адаптивного бустингу, наївного прогнозу та НС МПГП лінійного типу. Результати прогнозування концентрацій ПЗб АП досліджуваними методами графічно представлено на рисунку 3.11.

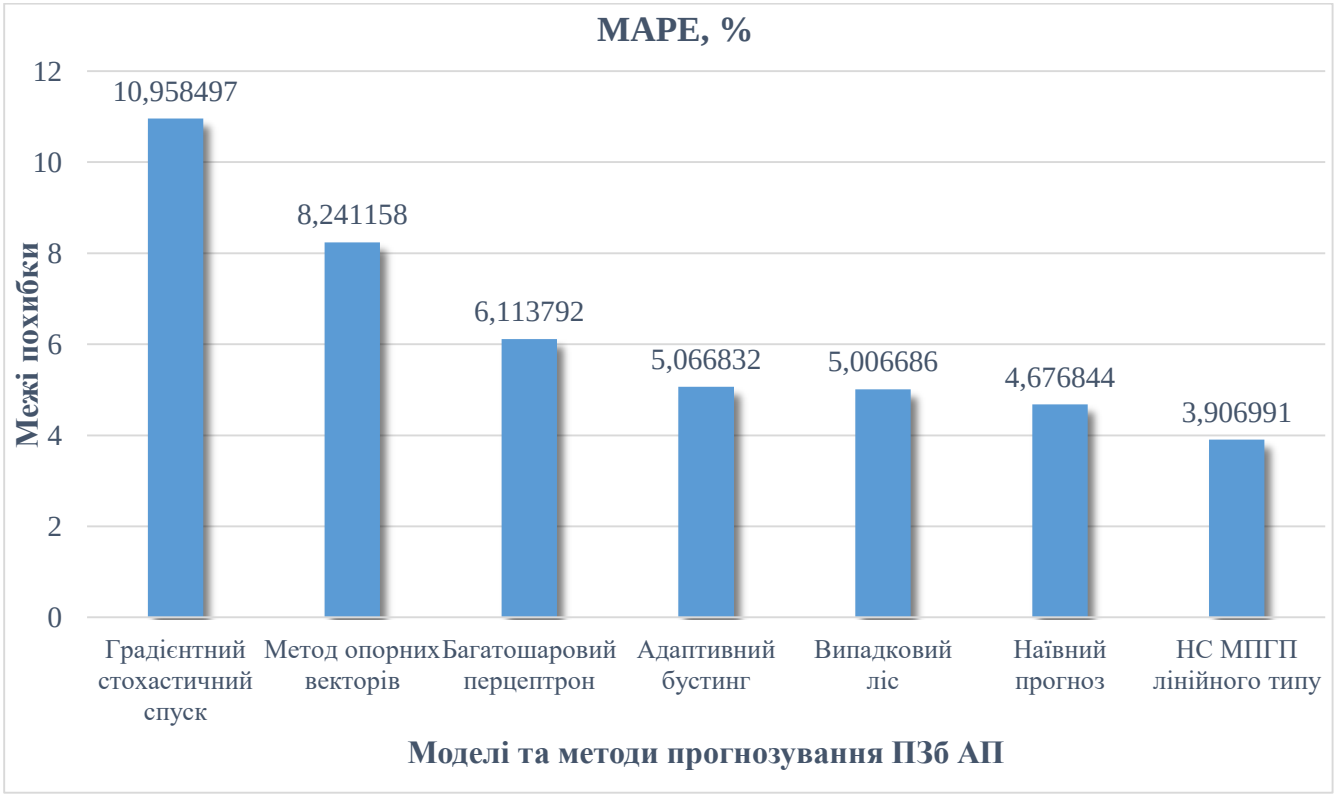


Рис. 3.11. Похибики прогнозування ПЗб атмосферного повітря, MAPE

З рискунку 3.11 випливає, що використання НС МПГП має найточніші результати однокровокого прогнозування серед інших досліджуваних моделей та методів прогнозування ПЗб АП. Тому наступним етапом реалізації короткотремінового прогнозування виконано багатокрокового прогнозування параметрів забруднення атмосферного повітря. Для прикладу, на рисунках 3.12а та 3.12б зображено результати багатокрокового прогнозування тренду забруднення атмосферного повітря шкідливою домішкою CO для часового вікна t_{iw} = 10 за домопогою НС МПГП лінійного типу.

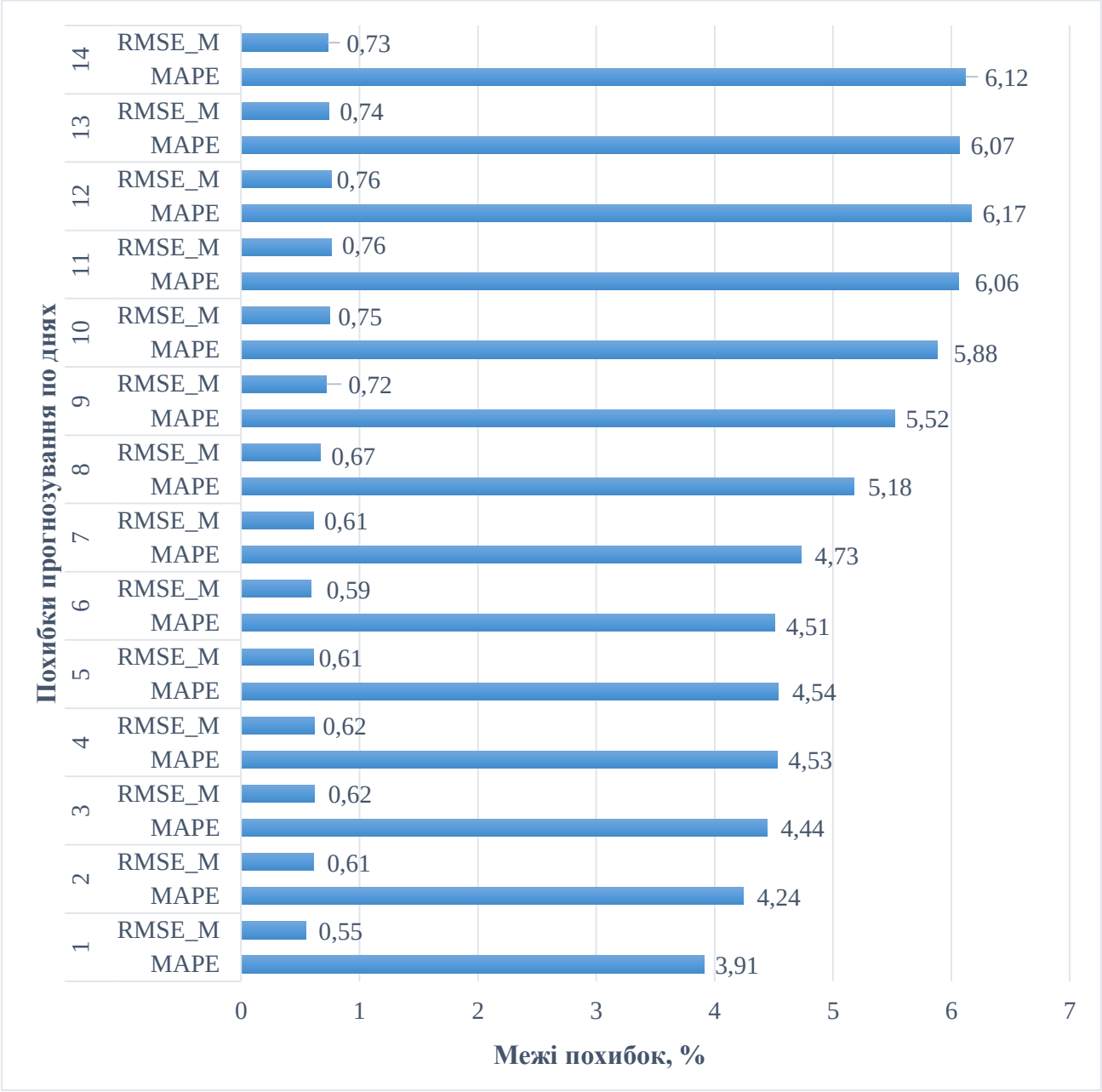


Рис. 3.12а. Похибки прогнозування тренду забруднення CO

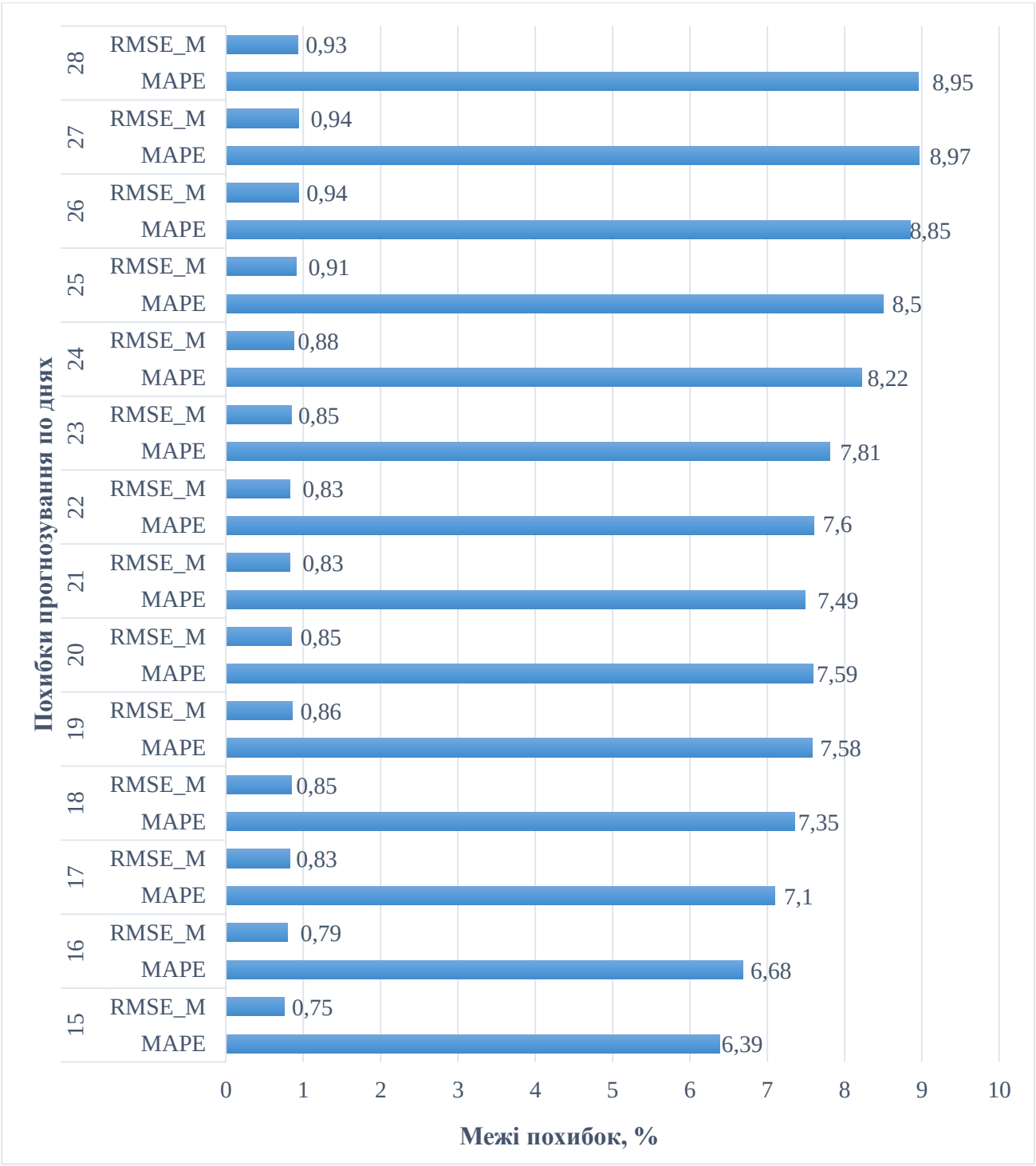


Рис. 3.12б. Похибки прогнозування тренду забруднення CO на другий тиждень

У дослідженні використано дані спостережень за викидами шкідливих домішок двічі на день. Досягнувши 29-го показника тренду забруднення атмосферного повітря шкідливою домішкою CO похибка прогнозування перевищила межу 10 %. Тому горизонтом прогнозування вибрано два тижні (28 показників, тобто 14 днів), що задовольняє умову короткотермінового прогнозування.

3.3.2.2. Результати часових затримок досліджуваних моделей та методів прогнозування концентацій діоксиду азоту

Результати швидкості прогнозування концентрацій параметрів забруднення атмосферого повітря графічно представлено на рисунку 3.13. у вигляді діаграми.

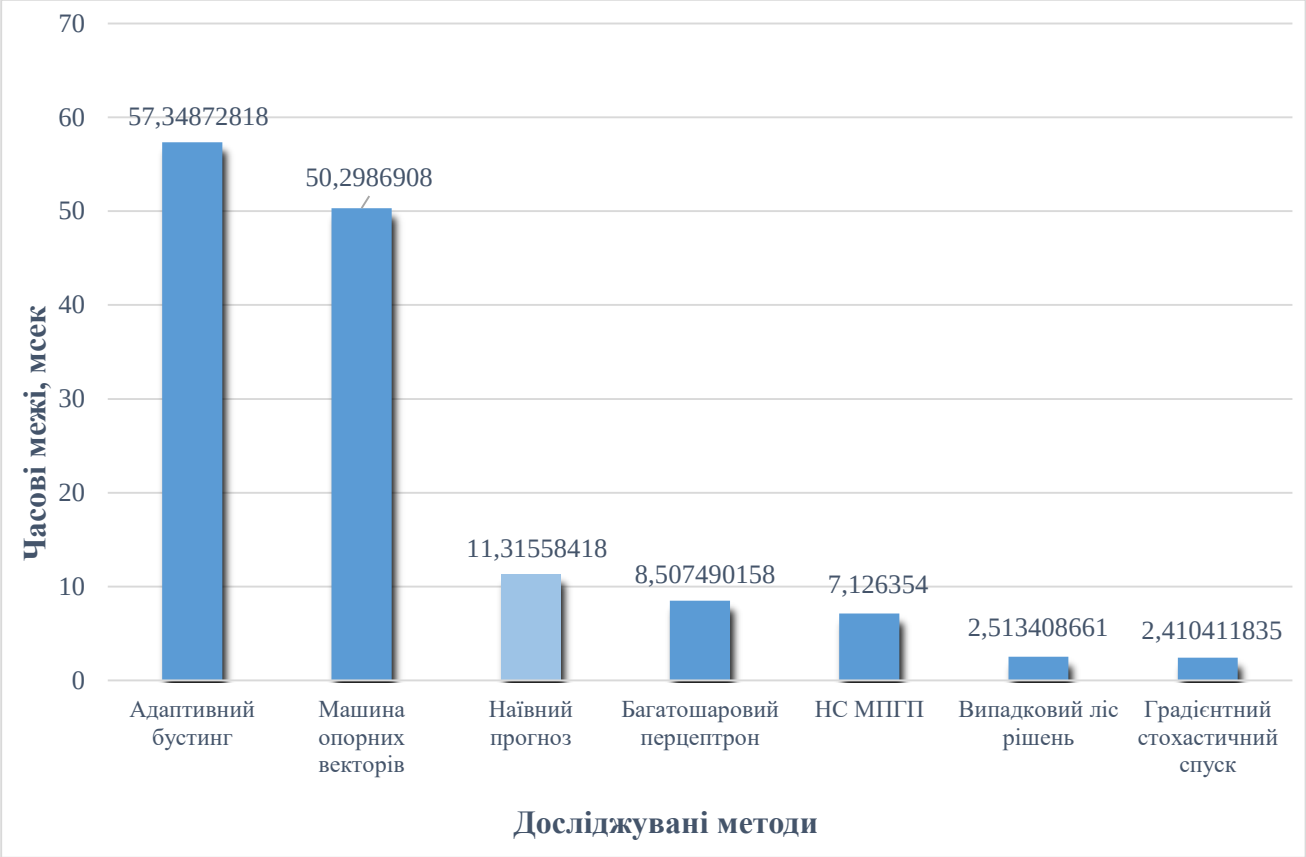


Рис. 3.13. Час застосування досліджуваних моделей та методів прогнозування параметрів забруднення атмосферного повітря

З рисунку 3.13 випливає, що під час використання НС МПГП з одним виходом затрачається більше часу в режимі застосування, ніж використаня дерева рішень чи стохастичного градієнтного спуску. В попередньому параграфі 3.3.2.1. досліджено, що використання НС МПГП має найточніші похибки прогнозування, та це єдиний метод, котрий має менші похибк за метод наївного прогнозу. Тому необхідним є розвиток методу пришвидшеного прогнозування параметрів забруднення повітряного середовища на мобільних пристроях та мікроконтролерах.

ВИСНОВКИ ДО РОЗДІЛУ 3

1. Розроблено метод короткотермінового прогнозування параметрів забруднення атмосферного повітря за допомогою комітету лінійної та нелінійної нейроподібних структур моделі послідовних геометричних перетврень для часткового коректування окремо додатних і від'ємних відхилень від точних значень. Наведено основні етапи та процедури реалізації методу.

2. Виконано порівняльну оцінку точності таких моделей та методів прогнозування концентрацій параметрів забруднення атмосферного повітря як багатошаровий перцептрон, дерево рішень, адаптивний бустинг, метод опорних векторів, НС МПГП, стохастичний градієнтний спуск та інші. Доведено, що найточнішим є НС МПГП, оскільки має найнижчі похибки застосування серед інших досліджуваних методів.

3. Встановлено, що похибка однокрокового прогнозування концентрацій оксиду азоту у атмосферному повітрі на основі НС МПГП становить 3,9 %, а горизонт прогнозування складає два тижні.

4. Проаналізовано метод пришвидшення зміни роздільної здатності шляхом нейромережевої ідентифікації коефіцієнтів ваг синаптичних зв'язків НС МПГП з багатьма виходами для задачі пришвидшення прогнозування параметрів забруднення атмосферного повітря на мобільних пристроях та мікроконтролерах.

5. Розвинуто досліджений метод на задачі прогнозування за рахунок використання НС МПГП не з багатьма виходами, а з одним. Також, розвиток полягає у застосуванні лінійних нейроподібних структур моделі послідовних геометричних перетворень з одним виходом та побудови на їх основі матриці коефіцієнтів не синаптичнх зв'язків, а матриці коефіцієнтів лінійних поліномів для прогнозування параметрів забруднення атмосферного повітря.

6. Встановлено, що час прогнозування концентрацій діоксиду азоту за допомогою методу побудови апроксимаційних поліномів шляхом ідентифікації їх коефіцієнтів за результатами навчання відповідних нейроподібних структур забезпечується зменшення затрат пам'яті під час прогнозування параметрів забруднення атмосферного повітря в 12,75 разів.

РОЗДІЛ 4. РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ ДЛЯ ПРОГНОЗУВАННЯ ПАРАМЕТРІВ ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ В УМОВАХ ПРОПУСКІВ У ДАНИХ

У розділі представлено розроблений програмний засіб з набором бібліотек для реалізації описаних у попередніх розділах розроблюваних, удосконалюваних та розвинених методів прогнозування викидів забруднюючих речовин у повітряне середовище, в тому числі в умовах пропущених концентрацій параметрів у даних моніторингу забруднення атмосферного повітря. Програмний засіб складається не лише зі згаданих методів, а й з функціоналу збору, предобробки та аналізу даних (наприклад масштабування вхідних векторів даних); комунікації розроблених фреймворків між собою; обрахунку різних типів похибок; додаткових методів опрацювання даних (наприклад, зчитування / запису даних з файлу) та бази даних, де виконується збереження зібраних даних та результатів досліджень.

4.1. Вибір технологій для розробки бібліотеки програмних засобів

Реалізація нейромережевих та нейроподібних алгоритмів заповнення пропусків у даних та прогнозування параметрів забруднення атмосферного повітря (описаних у попередніх розділах) здійснена мовою Python, котра є високорівневою мовою програмування загального призначення з мінімалістичним синтаксисом та низкою вільних бібліотек з доступними алгоритмами машинного навчання. Python підтримує паралельне програмування на рівні потоків з деякими особливостями за допомогою модуля threading та використовує GIL [158] для синхронізації між потоками, тому є можиливим прискорення виконання задачі прогнозування. Також Python підтримує паралельність на рівні процесів за допомогою модуля multiprocessing, який використовує прикладний програмний інтерфейс аналогічно модулю threading.

Python підтримує розподілене програмування за допомогою бібліотек. Для цього використовується бібліотека RPyC для віддалених викликів процедур, бібліотека Celery для виконання завдань на окремих машинах і об'єднання результатів. За допомогою декоратора singledispatch з модуля functools можна

створювати узагальнені функції. Отже, перевагами цієї мови програмування велика кількість вільних бібліотек, безкоштовне використання, компактний синтаксис.

У розробленому Python-фреймворку виконується налаштування параметрів реалізованих алгоритмів прогнозування концентрацій забруднення атмосферного повітря в тому числі в умовах пропусків у даних, їх навчання та застосування. Алгоритми розроблені використовуючи пакет Anaconda Python – повністю налаштоване середовище програмування з попередньо установленою низкою бібліотек різних версій.

В програмному проекті мовою Python виконується реалізація таких алгоритмів прогнозування параметрів забруднення атмосферного повітря: алгоритм на основі машини опорних векторів (SVR), алгоритм на основі лінійної регресії зі стохастичним градієнтним спуском (SGDr), Adaptive Boosting, алгоритми на основі дерева рішень та випадкового лісу, алгоритми на основі нейроподібної структури моделі послідовних геометричних перетворень (НС МПГП), алгоритм на основі багатошарового перцептрону, методи наївного прогнозу та заповнення пропусків середнім значенням.

Основним недоліком мови Python є відсутність підтримки багатопотоковості на рівні процесорів, що може збільшувати часові затримки прогнозування параметрів забруднення атмосферного повітря в реальному часі. Тому для реалізації розроблених методів підвищення точності та швидкодії методів прогнозування параметрів забруднення повітряного середовища в умовах пропусків у даних використано мову Java, котра підтримує багатопотоковість та є універсальною для програмування комп’ютерних пристроїв (мобільних телефонів, контролерів, планшетів і т.д.).

Серед безлічі спеціалізованих і універсальних мов програмування, що застосовуються в задачах штучного інтелекту мова Java займає особливе місце [159]. Проста за синтаксисом, незалежна від платформи вона набула великої популярності серед спеціалістів галузі інформаційних технологій. З моменту її появи створено багато спеціалізованих засобів для задач штучного інтелекту.

Серед них для навчання і підготовки професійних програмістів популярні такі середовища [160]: оболонка JESS для створення експертних систем на основі баз знань у виді правил, jFuzzyLogic для розв'язання задач користувача методами нечіткої логіки, NeurophStudio для створення нейромереж різної структури, JADE для створення інтелектуальних агентів. Окрім цього існує багато окремих бібліотек процедур, написаних на Java, які реалізують класичні алгоритми штучного інтелекту. Ці процедури легко інтегруються з різними проектами.

Таким чином внутрішня логіка розробленого програмного засобу складається з двох фреймворків: Python-фреймвоку та Java-фреймвоку. Обидва програмні фреймворки взаємодіють між собою для досягнення поставленого завдання підвищення точності та швидкодії прогнозування параметрів забруднення атмосферного повітря в умовах пропусків у даних моніторингу повітряного середовища під час використання на мобільних та автономних пристроях на основі мікроконтролерів.

Користувацький інтерфейс програмного засобу для прогнозування параметрів забруднення атмосферного повітря, зокрема в умовах пропусків у даних, на мобільних пристроях розроблено у відкритому інтегрованому середовищі розробки Android Studio. Це середовище містить в собі засоби для спрощення перевірки сумісності аплікації з різними версіями платформи Android та для проектування застосунків, сумісних з пристроями різної роздільності (планшети, мобільні пристрої, ком'ютерні пристрої, годинники тощо). Основними перевагами Android Studio є можливість використання вбудованих емуляторів мобільних пристроїв, що дозволило розробити програмний засіб для мобільних пристроїв не використовуючи реальні пристрої.

Таким чином, розроблено програмний засіб для прогнозування параметрів забруднення атмосферного повітря в умовах пропусків у даних, котрий складається з користувацького інтерфейсу та двох фреймворків, де виконується внутрішня та низькорівнева логіка. Всі дані, починаючи від вхідних і закінчуючи результатами заповнення та прогнозування зберігаються в базі даних MS SQL Express, котра є безкоштовною.

4.2. Загальна архітектура розробленого програмного засобу для прогнозування параметрів забруднення повітря на мобільних пристроях

Загальні етапи реалізації програмного засобу для прогнозування та заповнення пропущених концентрацій параметрів забруднення атмосферного повітря зображено на рисунку 4.2. у вигляді блок-схеми.

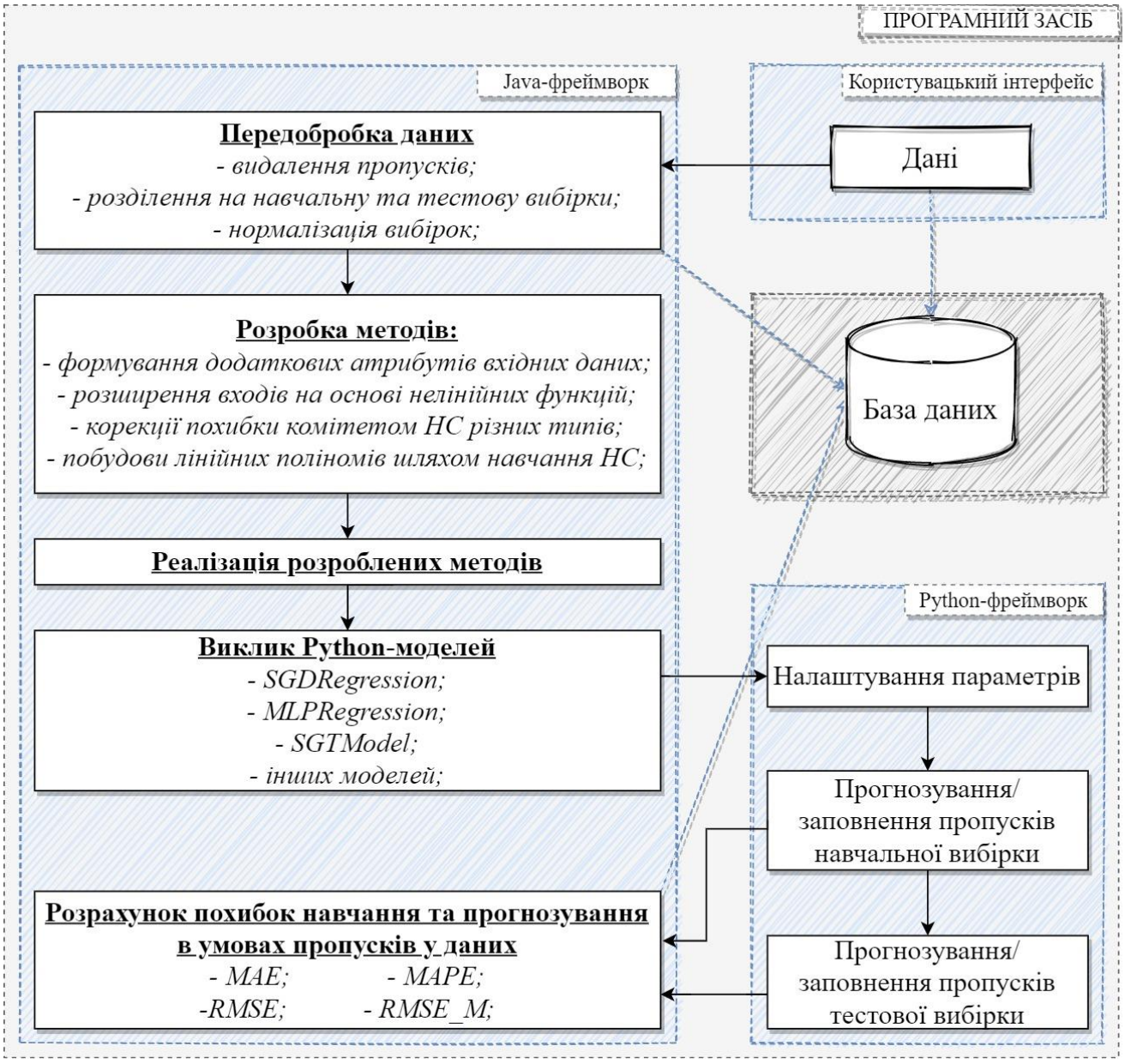


Рис. 4.2. Загальна архітектура розробленого програмного засобу

Отже, зображений на рисунку 4.2. програмний засіб для прогнозування та заповнення пропущених концентрацій параметрів забруднення атмосферного повітря на мобільних пристроях складається з користувацького інтерфейсу, двох фреймворків з низькорівневою логікою та бази даних для зберігання даних.

4.2.1. Опис розробленого Python-фреймворку для прогнозування параметрів забруднення повітря в умовах пропущених даних

Під час розробки нейромережевих та нейроподібних методів прогнозування в умовах пропущених концентрацій у параметрах забруднення повітряного середовища використано наступні сучасні, вільні у користуванні бібліотеки машинного навчання мови Python:

- Theano (дозволяє визначати, оптимізувати та оцінювати математичні вирази з використанням багатовимірних масивів);
- Keras (високорівневий прикладний програмний інтерфейс, побудований на вершині TensorFlow);
- Pandas (допомагає працювати з великою кількістю табличних даних);
- CUDA – Compute Unified Device Architecture (набір бібліотек, які дозволяють ефективно і швидко проводити обчислення на відеокартах NVidia).

Для розробки моделей прогнозування та заповнення пропущених концентрацій ПЗб АП використано бібліотеку sklearn, за допомогою якої реалізовано наступні моделі:

- *RandomForestRegressor* – модель регресії на основі випадкового лісу;
- *DecisionTreeRegressor* – модель регресії на основі дерева рішень;
- *AdaBoostRegressor* – модель регресії на основі адаптивного бустингу;
- *SGDRegressor* – модель лінійної регресії на основі стохастичного градієнтного спуску;
- *MLPRegressor* – модель регресії на основі багатошарового перцептрона;
- *SVR* – модель регресії на основі опорних векторів;
- *SGTM* – нейроподібна структура моделі послідовних геометричних перетворень;
- *AverageFill* – модель заповнення пропусків на основі усереднення;
- *RBFSample* – модель радіально-базисної функції;
- *NaivePredict* – модель наївного прогнозу.

4.2.2.1. Налаштування параметрів реалізованих моделей прогнозування

Кожна з реалізованих моделей прогнозування та заповнення пропущених концентрацій ПЗб атмосферного повітря включає в себе наступні загальні методи:

- для навчання: *fit (X_train, y_train)*;
- для пошуку R2 похибки: *score (X_train, y_train)*;
- для прогнозування чи заповнення пропусків: *predict (X_test)*.

Додатково кожна з моделей має власні параметри, котрі потрібно налаштувати для отримання ефективних результатів. Опишемо реалізацію налаштування параметрів моделей, однак детальний опис для кожної розробленої моделі прогнозування концентрацій показників забруднення повітряного середовища в умовах пропусків у даних наведено у Додатку Г.

Модель ***RandomForestRegressor*** використовує усереднення для підвищення точності прогнозування та містить в собі наступні параметри налаштування: число дерев лісу; критерій вимірювання якості розбиття; число максимальної глибини дерева; число мінімальної кількості зразків, необхідне для розділення внутрішнього вузла; число мінімальної кількості зразків у новостворених деревах; параметр, що контролює щільність розбиття; кількість функцій, які слід враховувати, шукаючи найкращий розділ; параметр, котрий вказує чи використовувати зразки для оцінки похибки узагальнення; вказівник кількості завдань, які потрібно виконати; та показник контролю процесу побудови дерев.

Модель ***DecisionTreeRegressor*** складається лише з одного дерева та містить схожі параметри налагодження до попередньої моделі *RandomForestRegressor*. Крім критерію вимірювання якості розбиття; числа максимальної глибини дерева; та числа мінімальної кількості зразків, необхідного для розділення внутрішнього вузла, до моделі DeciosionTreeRegressor входять наступні парметри: показник вибору стратегії розбиття на кожному вузлі; мінімальна зважена частка від загальної ваги; параметр, що задає найкращі вузли; параметр, що вказує чи вузол буде розщеплений; параметр, що встановлює поріг для ранньої зупинки росту дерева; та параметр, що використовується для зупинки росту дерева.

AdaBoostRegressor (метаоцінювач, який починається з установки регресора на початковий набір даних) містить в собі такі параметри: базовий оцінювач, з якого побудований підсилений ансамбль; максимальна кількість оцінювачів призупинення; коефіцієнт навчання; та параметр, що визначає функцію втрат, яку слід використовувати при оновленні ваг після кожної інтенсивної ітерації.

SGDRegressor - лінійна модель, оснащена мінімізацією регульованих емпіричних втрат за допомогою стохастичного градієнтного спуску: градієнт втрат оцінюється кожним зразком за один раз і модель оновлюється по дорозі зі зменшенням графіка міцності (він же і рівень навчання). Ця модель у розробленому Python-фреймворку має такі параметри: показни визначення функції втрат; константа, що підсилює регуляризацію; вказівник перехоплення оцінки; максимальна кількість пропусків у навчальних даних; критерій зупинки; вказівник змішування даних тренувань після кожної епохи; графік курсу навчання; початковий коефіцієнт навчання; коефіцієнт зворотного масштабування швидкості навчання; вказівник використання ранньої зупинки для припинення навчання; частка даних тренувань для перевірки раннього припинення; та показник кількості повторень, які не мають покращення.

Модель ***MLPRegressor*** приймає кілька входів із абсолютно різними значеннями та налаштовується наступними параметрами: кількість нейронів у і-му прихованому шарі; функція активації для прихованого шару; вирішувач для оптимізації ваги; розмір мініатюр для стохастичних оптимізаторів; вказівник оновлення ваги; показник ступеня зворотного масштабування; максимальна кількість ітерацій; показник перемішування зразків в кожній ітерації; вказівник толерантності до оптимізації; вказівник публікування повідомлення про прогрес; вказівник використання імпульсу Нестерова; вказівник припинення навчання, коли оцінка перевірки не покращується; вказівник частки даних тренувань для встановлення раннього припинення; експоненціальні швидкості занепаду для оцінок векторів першого та другого моменту; максимальна кількість епох без поліпшення; максимальна кількість викликів функцій.

Модель ***SVR*** містить в собі такі параметри налаштування: тип ядра, який буде використовуватися в алгоритмі; ступінь функції полінома ядра; коефіцієнт ядра (тип функції активації); допуск для зупинки критерію; параметр регуляризації; функція втрат; вказівник використання евристики скорочення; параметр визначення розміру кешу ядра (у МБ); вказівник включення докладного висновку та показник кількості ітерацій.

4.2.2.2. Налаштування параметрів НС МПГП

Налаштування НС моделі ***SGT*** складається з послідовності таких кроків:

Крок 1. Вибір обсягу вхідного шару рівного розмірності вхідного вектора.

Крок 2. Встановлення кількості вихідних нейронів, що визначається кількістю прогнозованих періодів.

Крок 3. Підбір кількості нейронів прихованого шару (надто велика кількість нейронів призводить до зростання похибки узагальнення).

Моделі ***SGTModel***, ***SGTModel_1*** та ***SGTModel_x_1*** відрізняються налаштованими параметрами для використання в режимах навчання та застосування. *SGTModel_1* та *SGTModel_x_1* мають налаштовані всі ті ж самі параметри, що і SGTModel, окрім кількості головних компонент, що повинні бути використані під час здійснення навчання алгоритму. Для моделі *SGTModel* кількість головних компонент дорівнює кількості вхідних векторів параметрів забруднення атмосферного повітря. Для моделі SGTModel_1 кількість головних компонент задається рівною одиниці, а модель *SGTModel_x_1* містить кількість головних компонент меншу на одиницю від кількості вхідних векторів даних. Всі три розроблені моделі відрізняються призначенням навчання та застосування. Модель *SGTModel* використовується для здійснення прогнозування параметрів забруднення повітряного середовища в умовах пропущених даних. Модель SGTModel_1 використовується для виділення тренду забруднення атмосферного повітря деяким параметром, а *SGTModel_x_1* здійснює згладжування вихідного вектора значень для лінеаризації поверхні відгуку, що підвищує точність прогнозування.

4.2.3. Опис розробленого Java-фреймворку для прогнозування параметрів забруднення повітря в умовах пропущених даних

Основним недоліком мови Python є відсутність підтримки багатопотоковості на рівні процесорів, що може збільшує часові затримки прогнозування параметрів забруднення атмосферного повітря на мобільних пристроях. Тому реалізація користувацького інтерфейсу та розроблених методів прогнозування параметрів забруднення повітряного середовища в умовах пропусків у даних виконана мовою Java, котра підтримує багатопотоковість та є універсальною для програмування комп'ютерних пристроїв (мобільних телефонів, контролерів, планшетів і т.д.).

Для реалізації частини програмного засобу у Java-фреймворку використано засоби автоматизації роботи проектів для управління (management) та складання (build) програм, так звані системи збору. Серед таких засобів розрізняють Gradle, Apache Maven, Ant та інші [170]. У роботі системою збору вибрано засіб Apache Maven, котрий описує структуру проекту в файлах мовою POM (підмножини XML). Уривок програмного коду створення залежностей за допомогою використання тегів XML наведено на рисунку 4.1.

```
<dependencies>
    <dependency>
        <groupId>net.sf.opencsv</groupId>
        <artifactId>opencsv</artifactId>
        <version>2.3</version>
    </dependency>
    <dependency>
        <groupId>org.apache.commons</groupId>
        <artifactId>commons-collections4</artifactId>
        <version>4.0</version>
    </dependency>
    <dependency>
        <groupId>org.python</groupId>
        <artifactId>jython-slim</artifactId>
        <version>2.7.2b3</version>
    </dependency>
</dependencies>
```

Рис. 4.1. Уривок програмного коду впровадження залежностей

Отже, для того, щоб програмний засіб коректно працював у проекті виконано впровадження залежностей бібліотек. До цих залежностей відносяться бібліотеки для виконання операцій над файлами, для взаємодії між фремворками різних мов, для використання баз даних та інші. Наприклад на рисунку 4.1. наведено уривок коду встановлення залежностей із зовнішньою бібліотекою *net.sf.opencsv* для відкривання файлів типу .csv та із бібліотекою допоміжних класів для роботи з колекціями *org.apache.commons*.

Частина програмний засобу для прогнозування концентрацій параметрів забруднення атмосферного повітря, Java-фреймворк, складається з двох модулів: користувацького інтерфейсу та внутрішньої логіки. Структура модуля внутрішньої логіки зображена на рисунку 4.2.

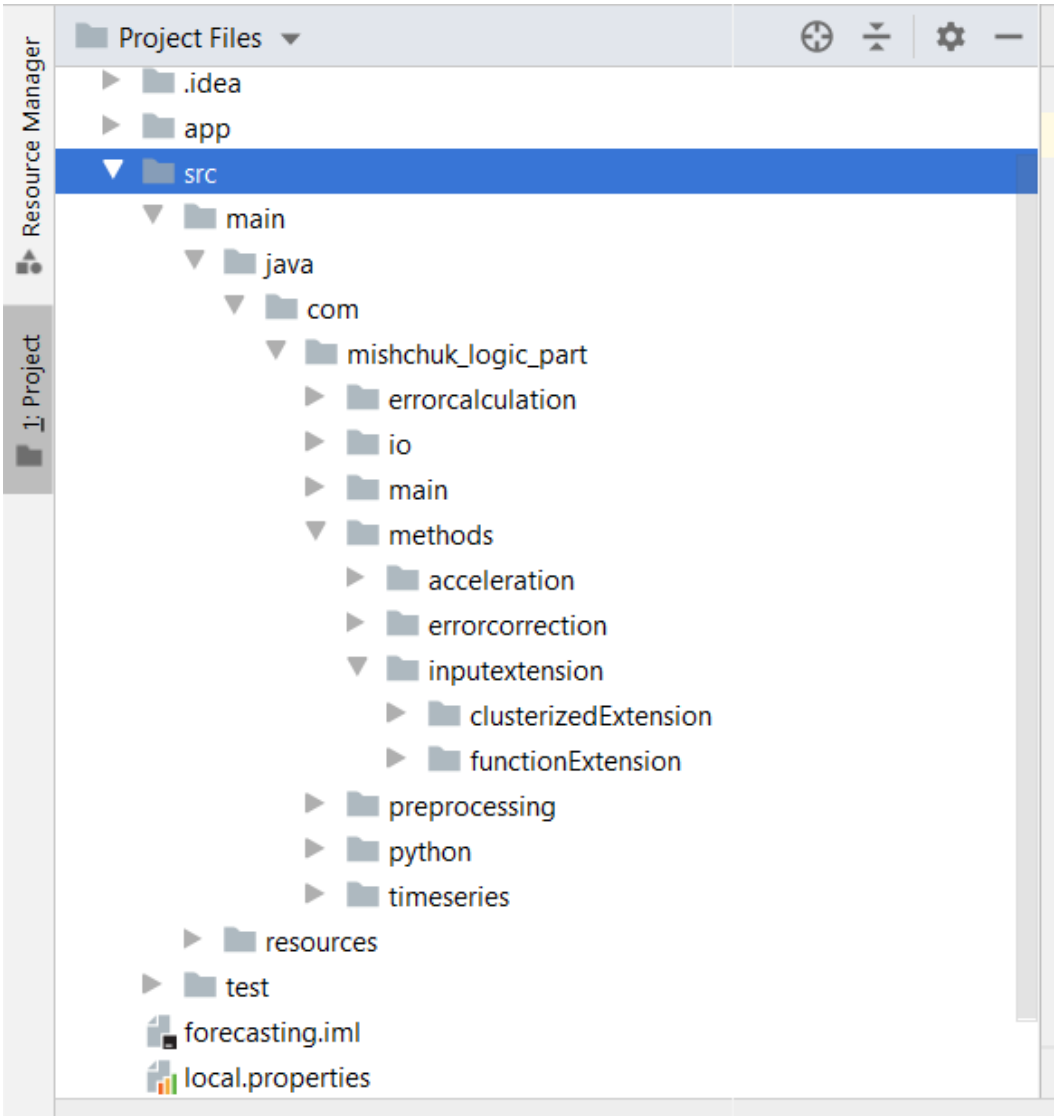


Рис. 4.2. Структура модуля внутрішньої логіки розробленого програмного засобу

З рисунку 4.2. випливає, що модуль внутрішньої логіки складається з наступних підмодулів: підмодуля обробки даних; підмодуля з набором розроблених методів; підмодуля для роботи з вхідними та вихідними даними; підмодуля взаємодії двох програмних проектів та підмодуля розрахунку точності та швидкодії реалізованих моделей та методів прогнозування параметрів забруднення атмосферного повітря, в тому числі в умовах пропусків у даних.

Кожен згаданий підмодуль складається з набору розроблених класів та методів. Наприклад, для можливості застосування методу формування додаткових атрибутів вхідних векторів на основі попередньої кластеризації векторів даних, розроблено наступні класи: Clusterization (клас для застосуваннякластеризації на досліджуваних даних моніторингу забруднення повітряного середовища), KNN (здійснення кластеризації векторів), Minimum (клас, де виконується пошук мінімальної відстані між веторами концентрацій параметрів забруднення атмосферного повітря, методи якого використовуються іншими класами), AbstractIO, Reader, Writer та інші. Ці класи містять методи зчитування даних з файлу формату «.csv», використовуючи метод readTable з класу Reader та writeTable з класу Writer для записування даних у файл, щоб вводити додаткові ознаки у вибірки даних.

Клас Clusterization містить:

- методи для застосування кластеризації навчальної та тестової вибірок концентрацій параметрів забруднення атмосферного повітря (clusterizeTrain та clusterizeTest);
- методи для визначення та генерування кластерів (getCluster та generateCluster);
- методи для обрахунків різних відстаней між вхідними атрибутами векторів даних (getDistance, getEuclidDistance та інші).

Для кращого розуміння набору класів та методів у розробленому модулі внутрішньої логіки програмного засобу варто розглнянути діаграму класів. Загальна діаграма усіх класів модуля внутрішньої логіки подана на рисунку 4.3.

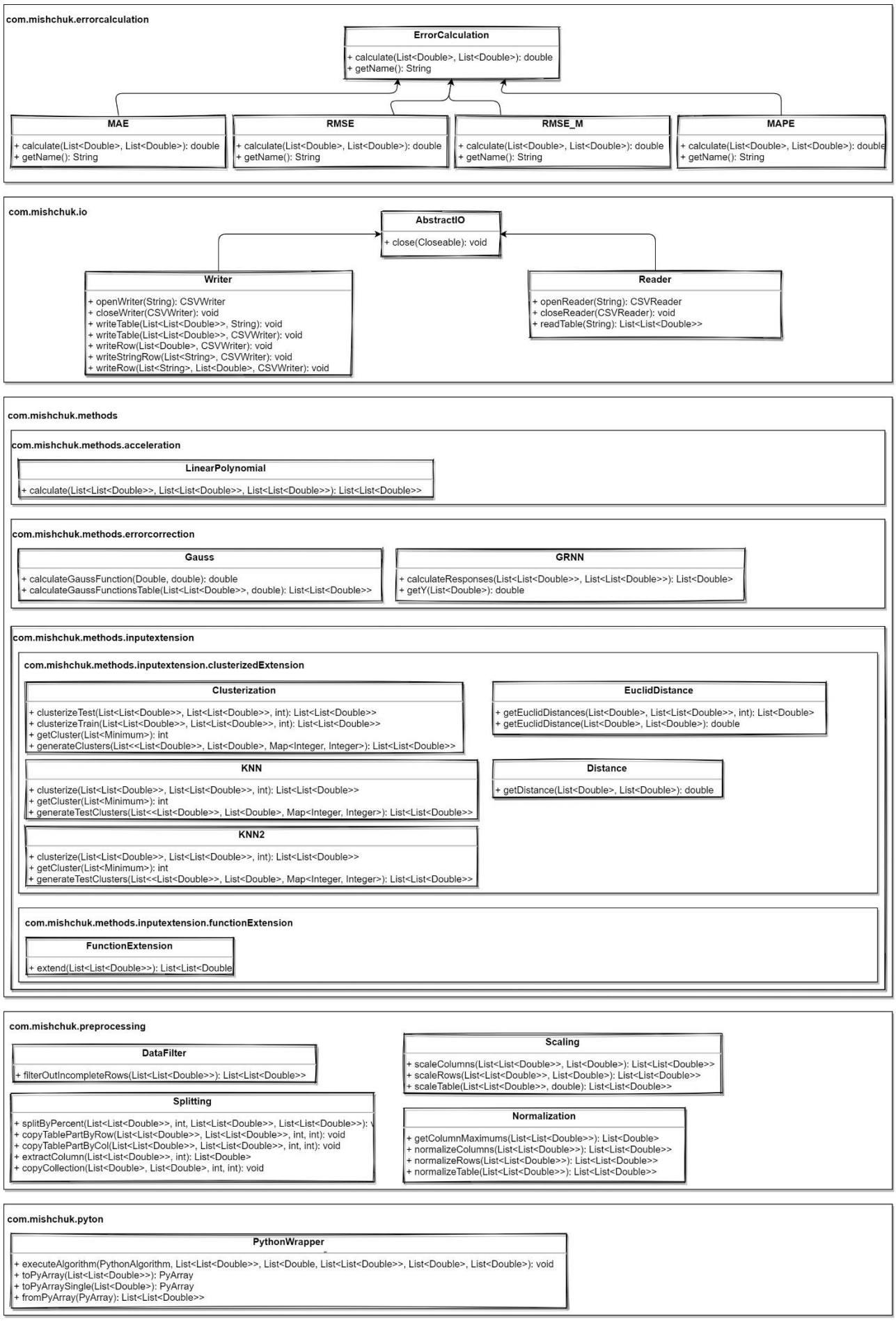


Рис. 4.3. Діаграма класів модуля внутрішньої логіки Java-фреймворку

4.2.3.1. Передобробка виміряних вибірок концентрацій параметрів забруднення атмосферного повітря

Підмодуль передобробки даних включає в себе такі класи (рис. 4.2. та 4.3.):

- *DataFilter.class* – клас, що виконує фільтрування вибірки вхідних даних;
- *Normalization.class* – клас, що нормалізує тренувальну та тестову матриці різними методами;
- *Scaling.class* – клас з методами виконання дій над таблицями;
- *Splitting.class* – клас, що містить методи розділення вхідної вибірки даних на навчальну та тестову.

DataFilter.class фільтруючи вхідну вибірку значень, видаляє вектори даних моніторингу атмосферного повітря, де є пропущені концентрації параметрів забруднення повітряного середовища.

Normalization.class розроблено для можливості виконання різних методів масштабування навчальної та тестової вибірок даних. Одним із методів масштабування є нормування до одиниці, котрий полягає у пошуку максимальних значень $max_{abs}(X_j)$ та $max_{abs}(Y)$ по стовпцях навчальної вибірки даних. Наступним кроком є масштабування всіх компонентів векторів тренувальної та тествої вибірок за формулами (4.1 - 4.3):

$$x_{i,j} = \frac{X_{i,j}}{max_{abs}(X_j)}, \tag{4.1}$$

$$x_{i,j} = \frac{X_{u,j}}{max_{abs}(X_j)}, \tag{4.2}$$

$$y_i = \frac{y_i}{max_{abs}(Y)}, \tag{4.3}$$

де $i = \overline{1,N}, j = \overline{1,M}$. .

Розроблений клас масштабування даних окрім методу нормування по стовпцях також містить методи нормування по рядках та всієї вибірки (крос-нормування). Уривок реалізації розробленого класу зображено на рисунку 4.4.

```
public class Normalization
{
    public static List<Double> getColumnMaximums(final List<List<Double>> input)
    {...}

    public static List<List<Double>> normalizeColumns(final List<List<Double>> input)
    {
        final List<Double> maxValues = getColumnMaximums(input);
        return Scaling.scaleColumns(input, maxValues);
    }

    public static List<List<Double>> normalizeRows(final List<List<Double>> input)
    {...}

    public static List<List<Double>> normalizeTable(final List<List<Double>> input)
    {...}
}
```

Рис. 4.4. Уривок програмного коду класу Normalization

Scaling.class розроблений для виконання дій над таблицями під час здійснення нормалізації вибірок даних за допомогою таки методів: scaleColumns, scaleRows та scaleTable (рис. 4.3.). Клас *Splitting.class* складається з розроблених методів для створення навчальних та тестових вибірок даних (рис. 4.5.).

```
public class Splitting
{
    public static void splitByPercent(final List<List<Double>> table, int percent,
                                      List<List<Double>> part1, List<List<Double>> part2)
    {...}

    public static void copyTablePartByRow(List<List<Double>> table,
                                          List<List<Double>> newTable, int offset, int rowCount)
    {...}

    public static void copyTablePartByCol(List<List<Double>> table,
                                          List<List<Double>> newTable, int offset, int colCount)
    {...}

    public static List<Double> extractColumn(final List<List<Double>> input, int columnNumber)
    {...}

    public static void copyCollection(final List<Double> source, final List<Double> dest,
                                      final int offset, final int count)
    {...}
}
```

Рис. 4.5. Уривок програмного коду класу Splitting

З рисунку 4.5. видно, що клас Splitting складається з методів розподілу вибірок: по відсотках, по рядках, по стовпцях та інших.

4.2.3.2. Підмодуль розроблених, удосконалених та розвинених методів

Підмодуль набору розроблених, удосконалених та розвинених методів підвищення точності та швидкодії прогнозування параметрів забруднення атмосферного повітря, в тому числі в умовах пропусків у даних моніторингу повітряного середовища складається з трьох компонент:

- *errorcorrection* – корекція похибки для підвищення точності прогнозування параметрів забруднення атмосферного повітря;
- *acceleration* – зменшення часових затримок застосування нейроподібних методів прогнозування/заповнення пропусків концентрацій параметрів забруднення повітряного середовища для використання на мобільних пристроях та мікроконтролерах;
- *inputextension* – розширення вхідних атрибутів векторів концентрацій параметрів забруднення атмосферного повітря;

Пакет *inputextension* включає в себе два підпакети – по одному для кожного методу розширення:

- *functionExtension* – реалізація методу розширення вхідних векторів даних параметрів забруднення атмосферного повітря, котрий має клас nonlinearExtension (удосконалення методу нелінійного розширення Йох-Хан Пао шляхом використання раціональних дробів);
- *clusterizedExtension* – реалізація методу формування додаткових атрибутів вхідних векторів даних шляхом попереднього виділення компактних множин точок (кластерів).

Підпакет clusterizedExtension містить в собі такі класи: Clusterization – клас з набором методів для реалізації різних видів виділення компактних множин точок (кластерів); DistanceCalculation – клас з набором методів для визначення різних видів відстаней між векторами ПЗб АП; ClusterizedExtension – клас з методами реалізації розширення атрибутів вхідних векторів концентрацій ПЗб АП. Детальний опис реалізації всіх розроблених методів наведено в Додатку Г.

4.2.3.3. Підмодуль взаємодії двох розроблених фреймворків

Підмодуль взаємодії двох програмних проектів містить в собі клас PythonWrapper з методів запуску моделей прогнозування та заповнення пропущених концентрацій параметрів забруднення повітря (рис. 4.6.).

```
public class PythonWrapper
{
    private static final String PYTHON_SCRIPT_NAME = "neural_networks.py";
    private static PythonInterpreter interpreter = new PythonInterpreter();

    public enum PythonAlgorithm
    {   ADAPTIVE_BOOSTING( pythonModelName: "Adaptive Boosting"),
        LINEAR_REGRESSION( pythonModelName: "Linear Regression"),
        DECISION_TREE( pythonModelName: "Decision Tree"),
        LINEAR_GRADIENT_DESCENT( pythonModelName: "Linear Gradient Descent"),
        MULTILAYER_PERCEPTRON( pythonModelName: "Multilayer Perceptron"),
        NATIVE_PREDICT( pythonModelName: "Naive Predict"),
        NEAREST_NEIGHBORS( pythonModelName: "Nearest Neighbors"),
        RANDOM_FOREST( pythonModelName: "Random Forest"),
        SUPPORT_VECTOR_REGRESSION( pythonModelName: "Support Vector Regression"),
        SGTM( pythonModelName: "Sequential Geometric Transformations Model"),
        RBF_SAMPLER( pythonModelName: "Radial Basis Function Sampler");

        private String pythonModelName;
        PythonAlgorithm(final String pythonModelName) { this.pythonModelName = pythonModelName; }

        public String getPythonModelName() { return pythonModelName; }  }

    public static void executeAlgorithm(final PythonAlgorithm algorithm, final List<List<Double>> trainX,
                                         final List<Double> trainY, final List<List<Double>> testX,
                                         final List<Double> predictedTrainY, final List<Double> predictedTestY)
    {...}

    private static PyArray toPyArray(final List<List<Double>> input)
    {...}
```

Рис. 4.6. Уривок програмного коду класу PythonWrapper

Методи підмодуля взаємодії двох фреймворків виконують перетворення даних у формат зрозумілий для мови Python, а після виконання заповнення пропусків та прогнозування перетворення отриманих даних назад у зрозумілий формат мови Java. Також, серед методів взаємодії Java-фреймворку з Python-фреймворком головним є метод executeAlgorithm, котрий виконує виклики реалізованих моделей прогнозування концентрацій параметрів забруднення повітряного середовища в умовах пропусків у даних.

4.2.3.4. Розрахунок точності реалізованих методів

Точність моделей та методів прогнозування оцінюється розрахунком прийнятих за стандарт: середніх значень абсолютних похибок (MAE), середніх значень відносних похибок (MAPE) та середньоквадратичних похибок (RMSE) за формулами (4.4 – 4.6) [161]:

$$MAE = \frac{1}{n}\sum_{j=1}^{n}\left|y_j' - y_j\right| \tag{4.4}$$

$$RMSE = \sqrt{\sum_{j=1}^{n}\left(y_j' - y_j\right)^2} \tag{4.5}$$

$$MAPE = \frac{1}{n}\sum_{j=1}^{n}\left|\frac{y_j' - y_j}{y_j}\right| \cdot 100 \tag{4.6}$$

де y'_j – спрогнозоване значення, а y_j – реальне значення з тестової вибірки даних моніторингу забруднення атмосферного повітря.

Приклад реалізації розрахунку середнього значення абсолютної похибки (MAE) для навчання та застосування зображено на рисунку 4.7.

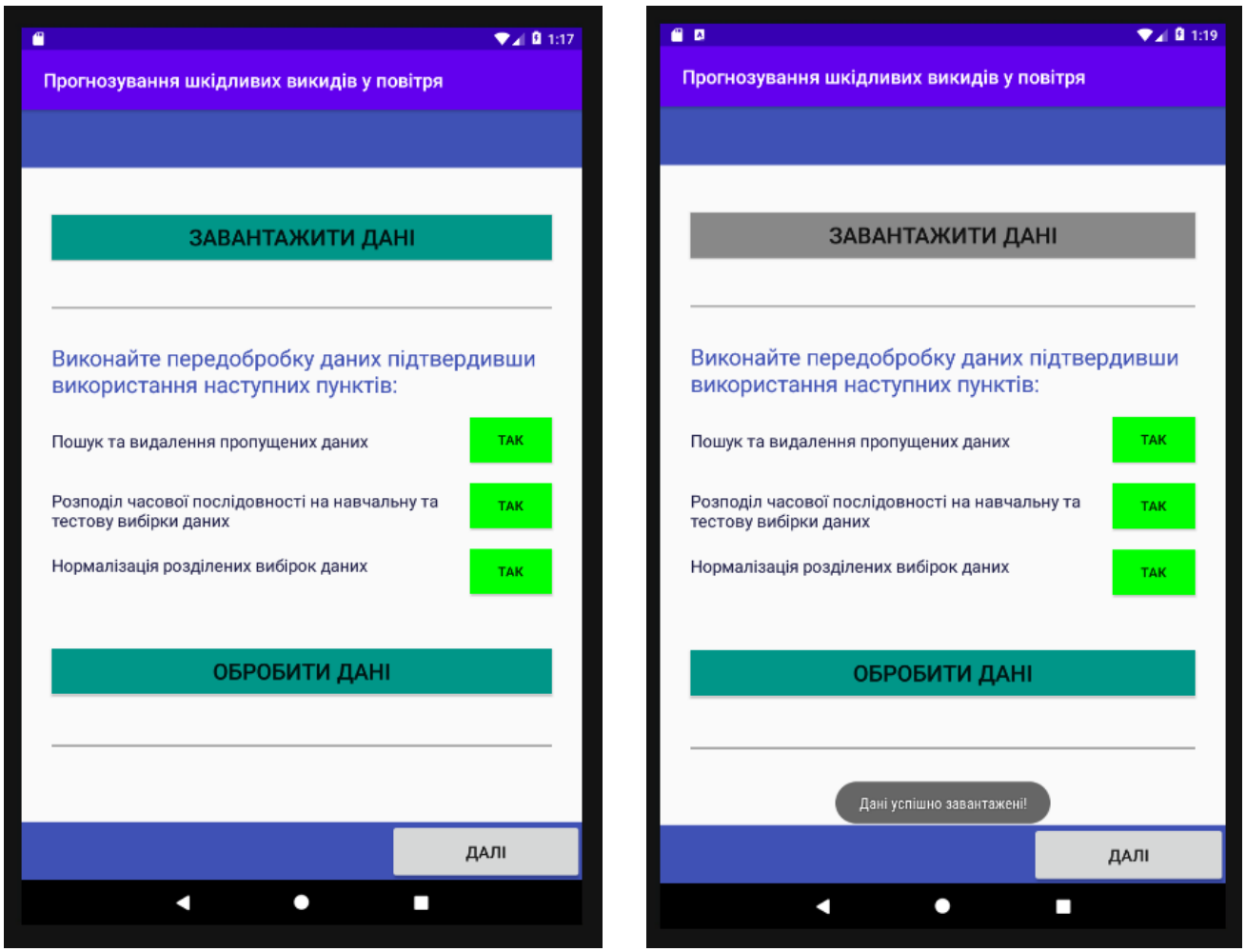
```
public class MAE implements ErrorCalculation
{
    @Override
    public double calculate(final List<Double> actual, final List<Double> predicted)
    {   double sum = 0.0d;
        for(int i = 0; i < actual.size(); i++)
        {   sum += Math.abs(actual.get(i) - predicted.get(i));      }
        return sum / actual.size(); }

    @Override
    public String getName() { return "MAE"; }
}
```

Рис 4.7. Уривок програмного коду реалізації підмодуля розрахунку похибок

На основі розрахованих похибок прогнозування в умовах пропущених концентрацій параметрів забруднення атмосферного повітря визначено який нейромережевий чи нейроподібний метод результує з найточнішими результатами. Експериментально доведено, що методи на основі НС МПГП показують точніші та швидші результати прогнозування та заповнення пропусків у даних за інші нейромережеві та нейроподібні моделі та методи.

4.2.4. Опис користувацького інтерфейсу розробленого програмного засобу

Розроблений програмний засіб для прогнозування параметрів забруднення атмосферного повітря на мобільних пристроях окрім внутрішньої логіки має користувацький інтерфейс. Отже, об’єднавши всі модулі програмного засобу, розроблено мобільну аплікацію під назвою «Прогнозування шкідливих викидів у повітря». При відкриванні розробленої мобільної аплікації на першому екрані є функції завантаження та обробки даних. Приклад успішного завантаження даних зображено на рисунку 4.8.

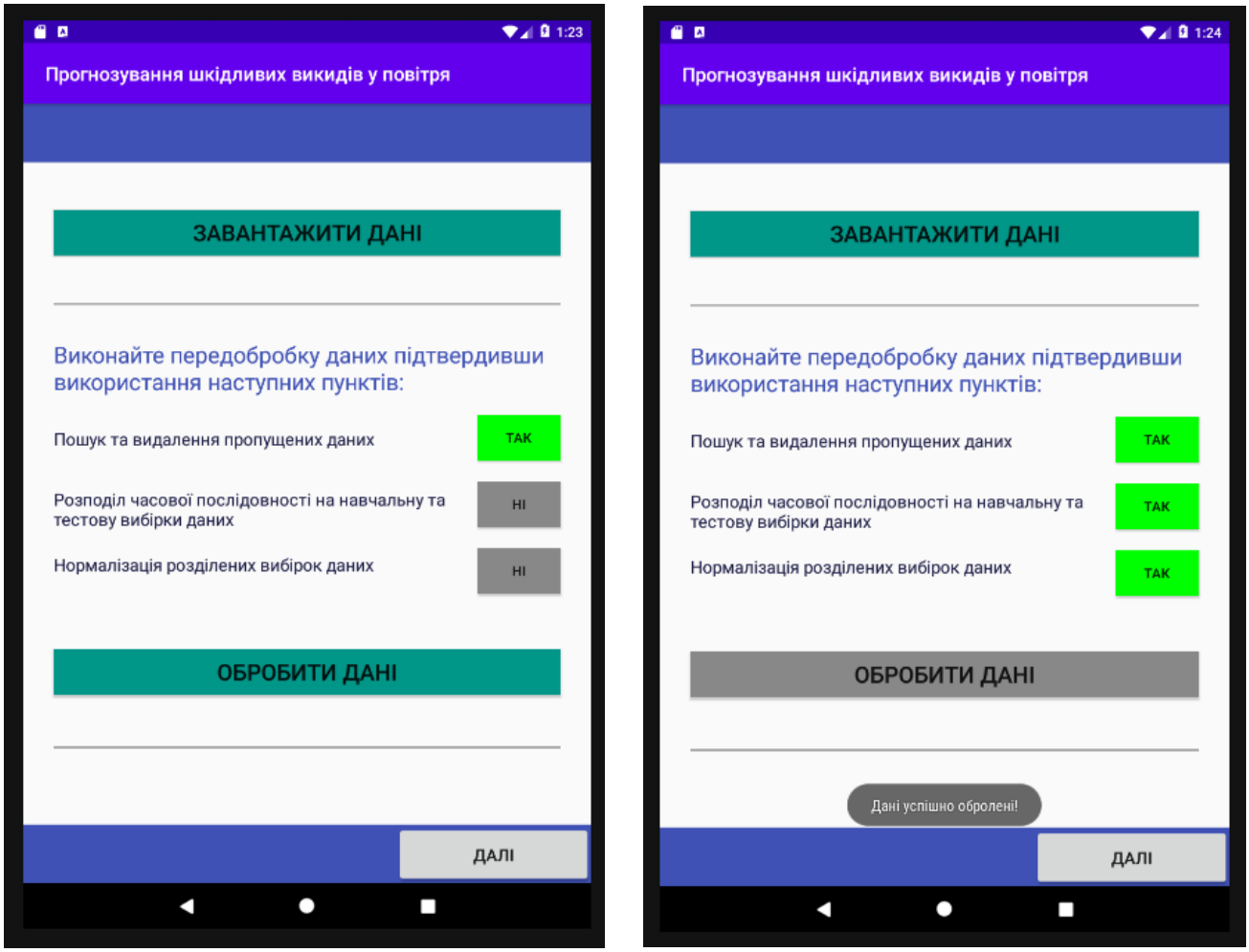


а) б)

Рис. 4.8. Приклад виконання завантаження даних

З рисунку 4.8. видно, що після виконання завантаження даних функція «завантажити дані» стає неактивною та залишається такою до переходу на наступний екран (оскільки враховується можливість повернення користувача для завантаження нових даних у разі помилки під час першого завантаження).

Після завантаження даних відбувається їх обробка. Користувач має можливість вибору процедур обробки даних залежно від кінцевих цілей. Список доступних процедур обробки завантажених даних зображено на рисунку 4.9.

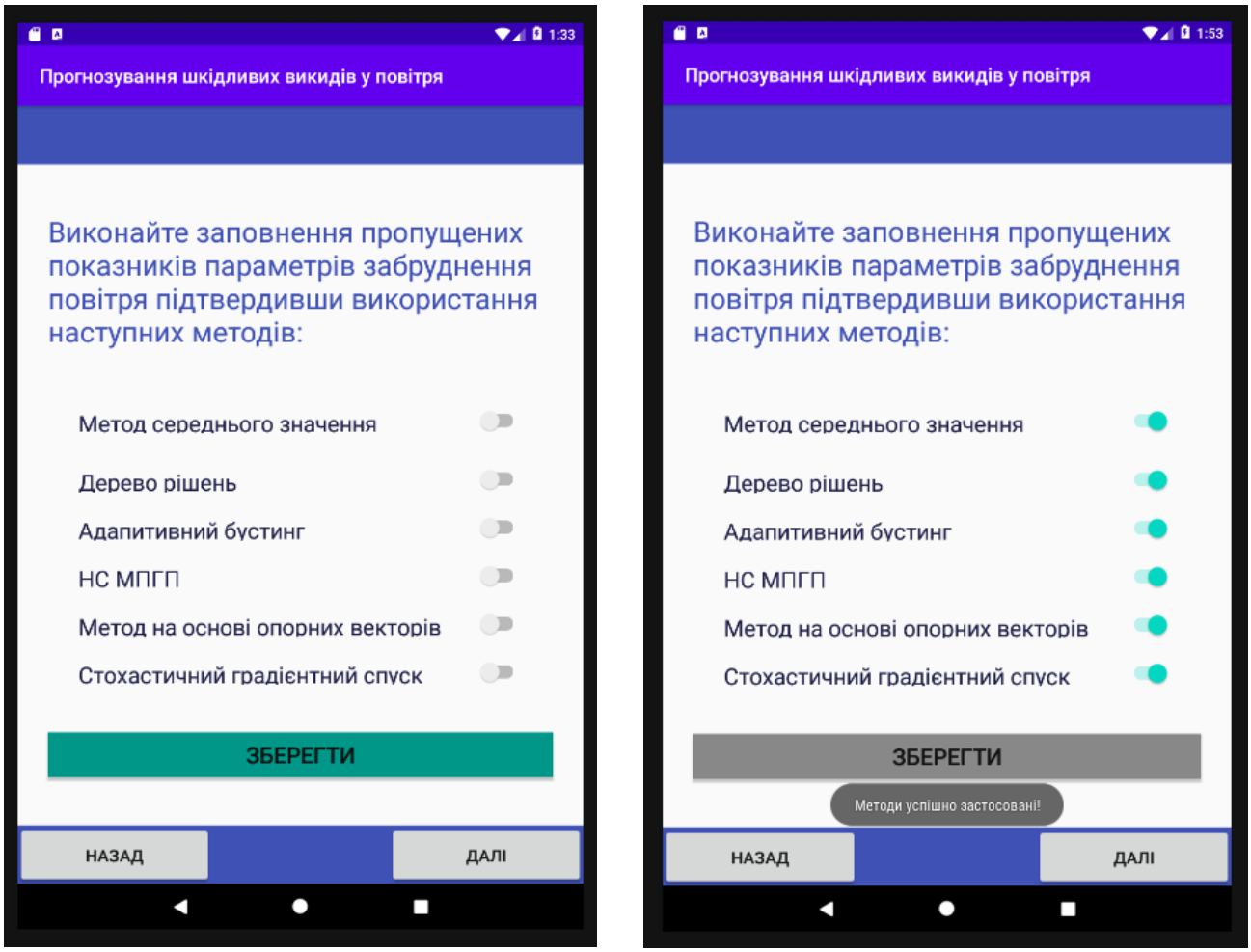


а) б)

Рис. 4.9. Приклад обробки даних

На рисунку 4.9а. зображено приклад успішної обробки даних підтвердивши використання всіх процедур передобробки. Користувач може окремо підтвердити чи відмінити виконання пошуку та видалення пропущених даних та виконання нормалізації розділених вибірок. Однак, у випадку відміни процедури розподілу завантажених даних на навчальну та тренувальну – нормалізація розділених вибірок також відміниться автоматично. Слід зазначити, що після обох етапів: завантаження та обробки даних – виконується їх зберігання у базі даних.

Коли дані успішно завантажені та оброблені відбувається перехід на наступний екран для виконання заповнення пропущених концентрацій ПЗб АП. Користувач повинен вибрати необхідний набір методів для заповнення пропусків. Перелік наявних методів зображено на рисунку 4.10.



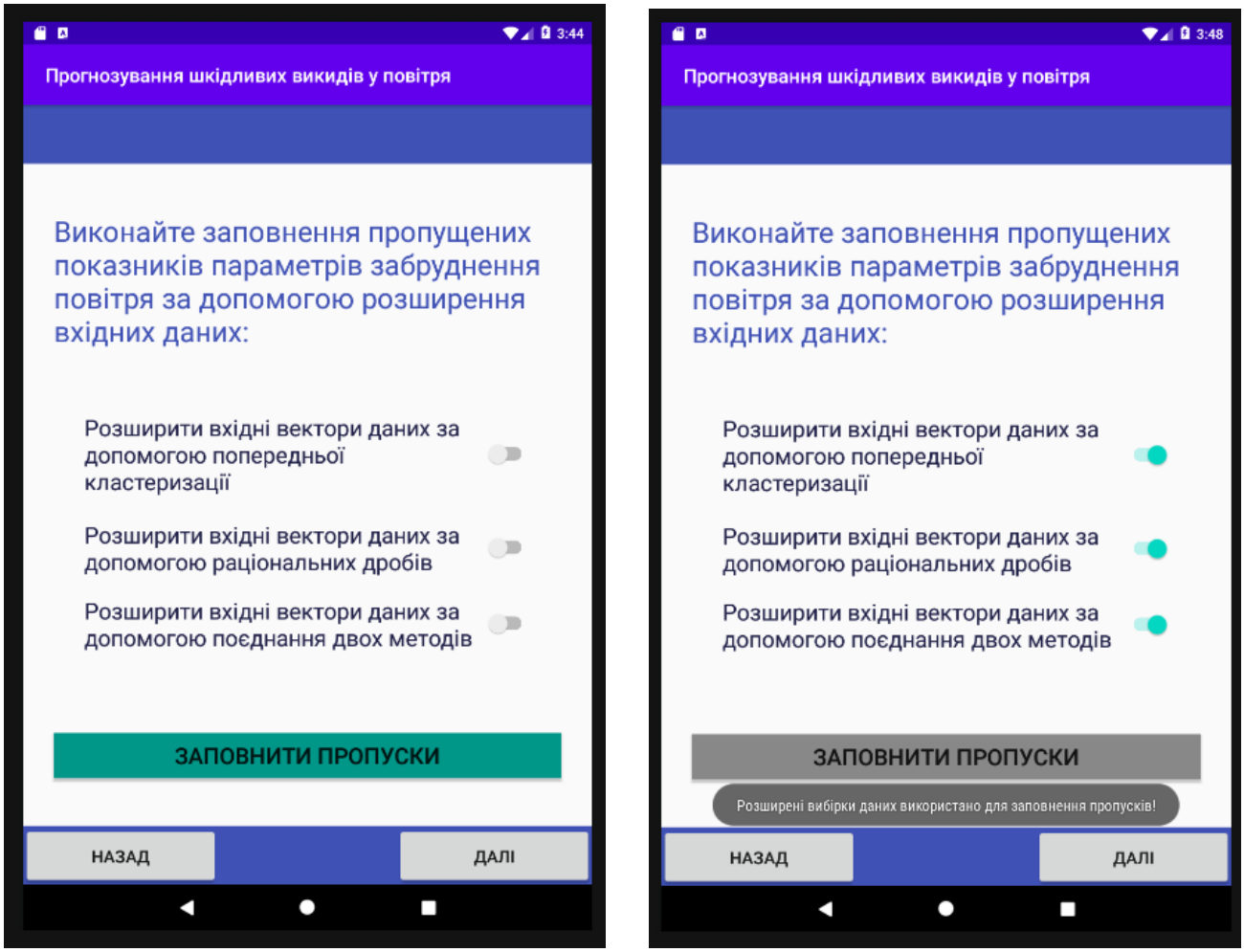
а) б)

Рис. 4.10. Виконання заповнення пропущених концентрацій ПЗб АП на завантажених вхідних даних

Якщо користувач не вибрав жодного методу, то на екрані з’являється випливаюче вікно з підтвердженням чи відміною етапу заповнення пропусків у даних моніторингу повітряного середовища. У випадку відміни – відбувається перехід на наступний етап та автоматично відкривається екран прогнозування параметрів забруднення АП. У випадку підтвердження – користувач повинен вибрати вибрати хоча б один із запропонованих методів.

Далі виконується застосування вибраних методів виконується заповнення пропущених концентрацій ПЗб АП. Отримані повні вибірки векторів даних моніторингу АП зберігаються у базі даних для подальшого використання та аналізу.

Наступним етапом є заповнення пропусків з використанням розширених даних за допомогою розроблених та удосконалених методів (рис.4.11.).

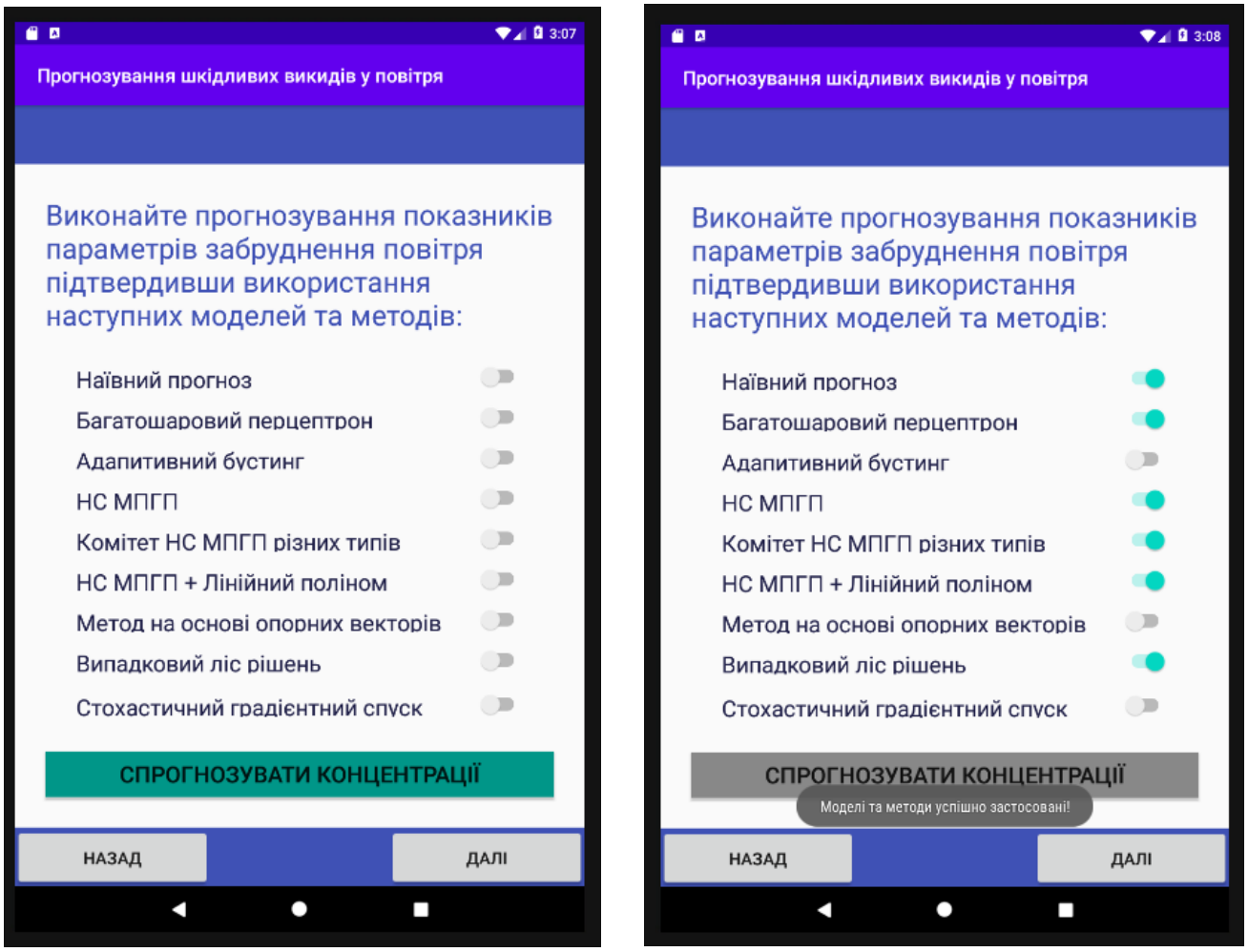


а) б)

Рис. 4.11. Виконання заповнення пропущених концентрацій ПЗб АП на розширених вхідних даних

Серед методів розширення користувач може вибрати метод розширення вхідних векторів даних за допомогою: попередньої кластеризації, раціональних дробів чи комбінацію обох методів. Якщо вибрано третій метод комбінування розширення, тоді використання двох попередніх методів підтверджується

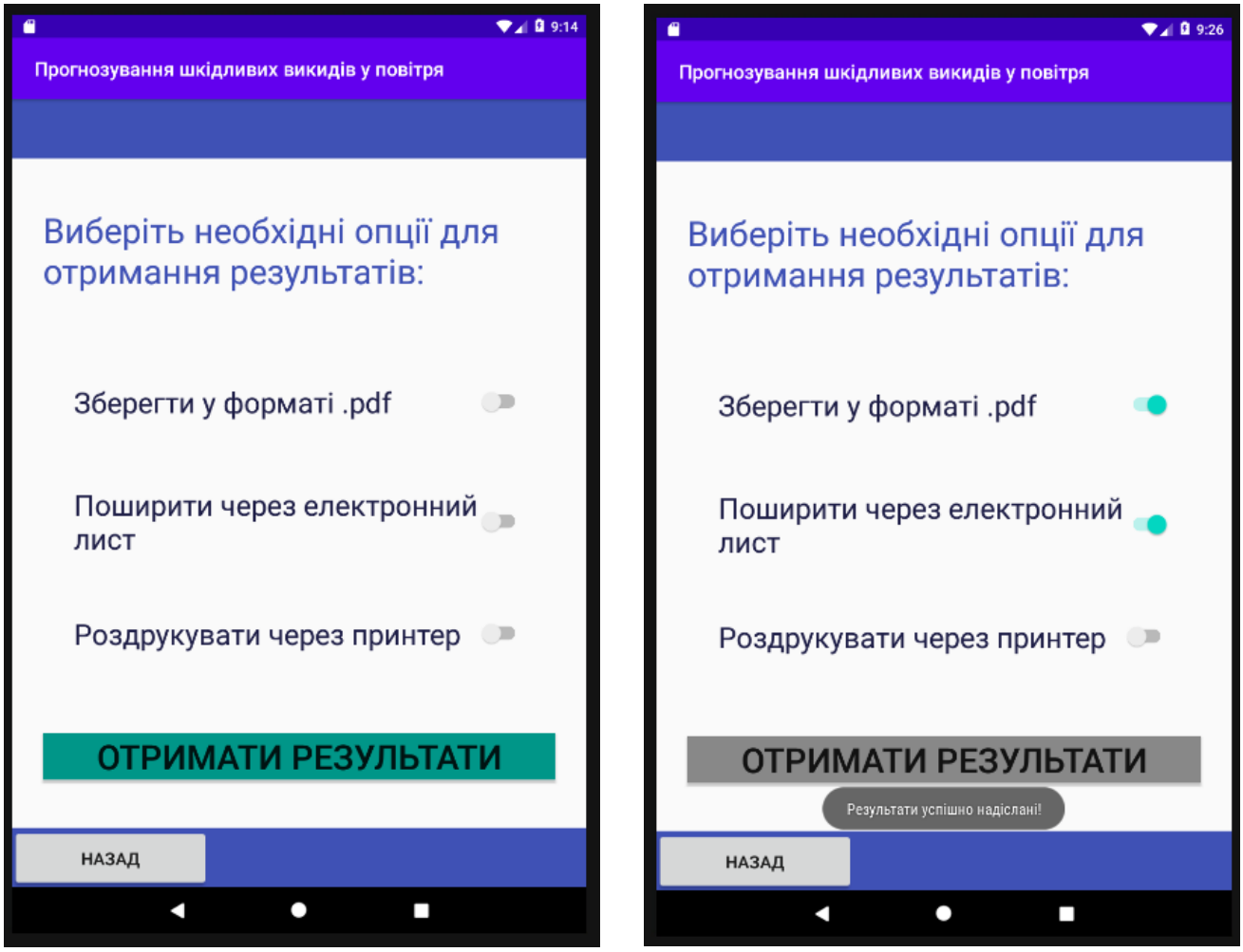
автоматично через технічні причини отримання даних з БД. Після вибору методів розширення, користувач виконує заповнення пропусків на розширених вибірках концентрацій ПЗб АП. Тоді на екрані з'являється випливаюче повідомлення про успішно виконане заповнення пропусків із використанням розширених даних, після чого користувач перенаправляється на екран прогнозування концентрацій ПЗб АП (рис. 4.12.).



а) б)

Рис. 4.12. Виконання прогнозування концентрацій ПЗб АП

На екрані виконання етапу прогнозування параметрів забруднення атмосферного повітря також (як і на попередніх екранах) є опція вибору набору моделей та методів для виконання поставленого завдання. Після підтвердження вибору необхідних моделей та методів, користувач виконує прогнозування концентрацій ПЗб АП та переходить на екран отримання результатів (рис. 4.13.).



а) б)

Рис. 4.13. Етап отримання результатів прогнозування ПЗб АП

Останнім екраном є отримання результатів прогнозування параметрів забруднення АП, де можна вибрати один з трьох варіантів: зберегти у файлі .pdf, поширити через електронний лист чи роздрукувати через принтер. Також є можливість поєднати два з трьох чи вибрати одразу всі варіанти отримання результатів. Після підтвердження отримання результатів вибраними способами користувач звіт про виконані етапи прогнозування. У звіті є опис початкових завантажених даних, їх аналіз та розширення. Також є результати виконання прогнозування та окремо заповнення пропусків у даних моніторингу забруднення атмосферного повітря. Зміст звіту залежить від того, які параметри, процедури, моделі чи методи були використані під час виконання мобільної аплікації.

4.3. Практична апробація розробленого програмного засобу

4.3.1. Реалізація та оцінка ефективності методів заповнення пропущених компонент параметрів забруднення атмосферного повітря

Реалізація методів заповнення пропущених концентрацій ПЗб АП складається з використання таких методів: зчитування даних; їх фільтрування (видалення пропусків); розбиття вибірки даних та навчальну та тестову; нормалізація обох вибірок; кластеризації векторів навчальної вибірки та інших. Перелік необхідних методів зображено в уривку коду на рисунку 4.14.

```
final List<List<Double>> input = Reader.readTable(inputPath);
final List<List<Double>> filteredInput = DataFilter.filterOutIncompleteRows(input);
final List<List<Double>> train = new ArrayList<>();
final List<List<Double>> test = new ArrayList<>();
Splitting.splitByPercent(input, 70, train, test);
final List<Double> trainMaximums = Normalization.getColumnMaximums(train);
final List<List<Double>> normalizedTrain = Scaling.scaleColumns(train, trainMaximums);
final List<List<Double>> normalizedTest = Scaling.scaleColumns(test, trainMaximums);
final List<List<Double>> trainX = new ArrayList<>();
Splitting.copyTablePartByCol(normalizedTrain, trainX, 0, train.get(0).size() - 1 - 1);
final List<Double> trainY = Splitting.extractColumn(normalizedTrain, train.get(0).size() - 1);
final List<List<Double>> testX = new ArrayList<>();
Splitting.copyTablePartByCol(normalizedTest, testX, 0, test.get(0).size() - 1 - 1);
final List<Double> testY = Splitting.extractColumn(normalizedTest, test.get(0).size() - 1);
final List<List<Double>> clusterizedTrain = Clusterization.clusterizeTrain(trainX, normalizedTest, 25);
final List<List<Double>> clusterizedTest = Clusterization.clusterizeTest(trainX, normalizedTest, 25);
final CSVWriter outputWriter = Writer.openWriter(outputPath);
final List<String> header = new ArrayList<>();
header.add("");
for(final ErrorCalculation ec : errorCalculationMethods)
{   header.add(ec.getName() + " Train");
    header.add(ec.getName() + " Test");
}
Writer.writeStringRow(header, outputWriter);
for(final PythonWrapper.PythonAlgorithm algorithm : PythonWrapper.PythonAlgorithm.values())
{   final List<Double> predictedTrainY = new ArrayList<>();
    final List<Double> predictedTestY = new ArrayList<>();
    PythonWrapper.executeAlgorithm(algorithm, clusterizedTrain, trainY, clusterizedTest, predictedTrainY, predictedTestY);
    final List<Double> row = new ArrayList<>();
    for(final ErrorCalculation ec : errorCalculationMethods)
    {   row.add(ec.calculate(trainY, predictedTrainY));
        row.add(ec.calculate(testY, predictedTestY));
    }
    Writer.writeRow(Collections.singletonList(algorithm.getPythonModelName()), row, outputWriter);
}
Writer.close(outputWriter);
```

Рис. 4.14. Уривок програмного коду реалізації методів заповнення пропусків на основі попереднього виділення кластерів

Для кращого розуміння розроблено методу заповнення пропущених концентрацій параметрів забруднення атмосферного повітря етапи його реалізації зображено на рисунку 4.15. у вигляді блок-схеми.

Рис. 4.15. Метод заповнення пропущених концентрацій ПЗб АП

Отже, заповнення пропущених концентрацій параметрів забруднення атмосферного повітря полягає у використанні розширених вхідних векторів даних для різних методів відновлення пропусків.

4.3.1.1. Результати заповнення пропусків удосконаленим методом Йох-Хан Пао шляхом використання раціональних дробів

У розділі досліджено удосконалений метод розширення вхідних векторів даних Йох-Хан Пао під час застосування різних методів заповнення пропусків на початкових та розширених даних вуглекислого газу та діоксиду азоту. Частину результатів досліджень наведено у таблиці 4.1.

Таблиця 4.1.

Похибки заповнення пропущених концентрацій оксиду карбону

	Похибки	Дані	SGDr	SVR	Adaptive Boosting	НС МПГП
CO	**MAE, test**	початкові	0,406174453	0,329298561	0,45938908	0,305470114
		розширені	0,388782	0,354488	0,458912	0,299855
	MAPE, test	початкові	30,68429514	25,57425911	42,15117683	22,38388325
		розширені	27,34642	26,10527	42,11985	22,00313
	RMSE, test	початкові	0,60432209	0,497646905	0,606752857	0,490839372
		розширені	0,579292	0,521056	0,605426	0,486203
	RMSE_M, test	початкові	5,078336892	4,181906761	5,098763506	4,124700607
		розширені	4,867998	4,378624	5,087614	4,08574

Наведені в табл. 4.1. дані у вигляді діаграм зображено на рис. 4.16а. - 4.16г.

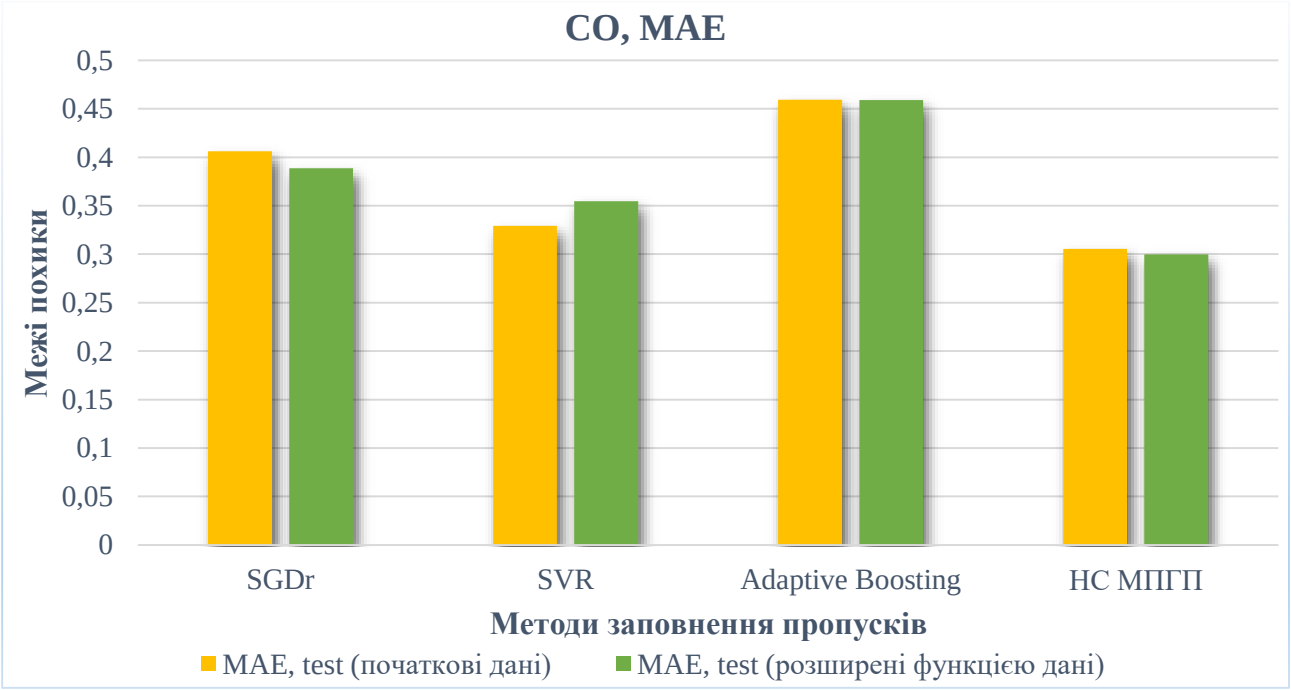


Рис. 4.16а. Тестові похибки заповнення пропусків вуглекислого газу, MAE

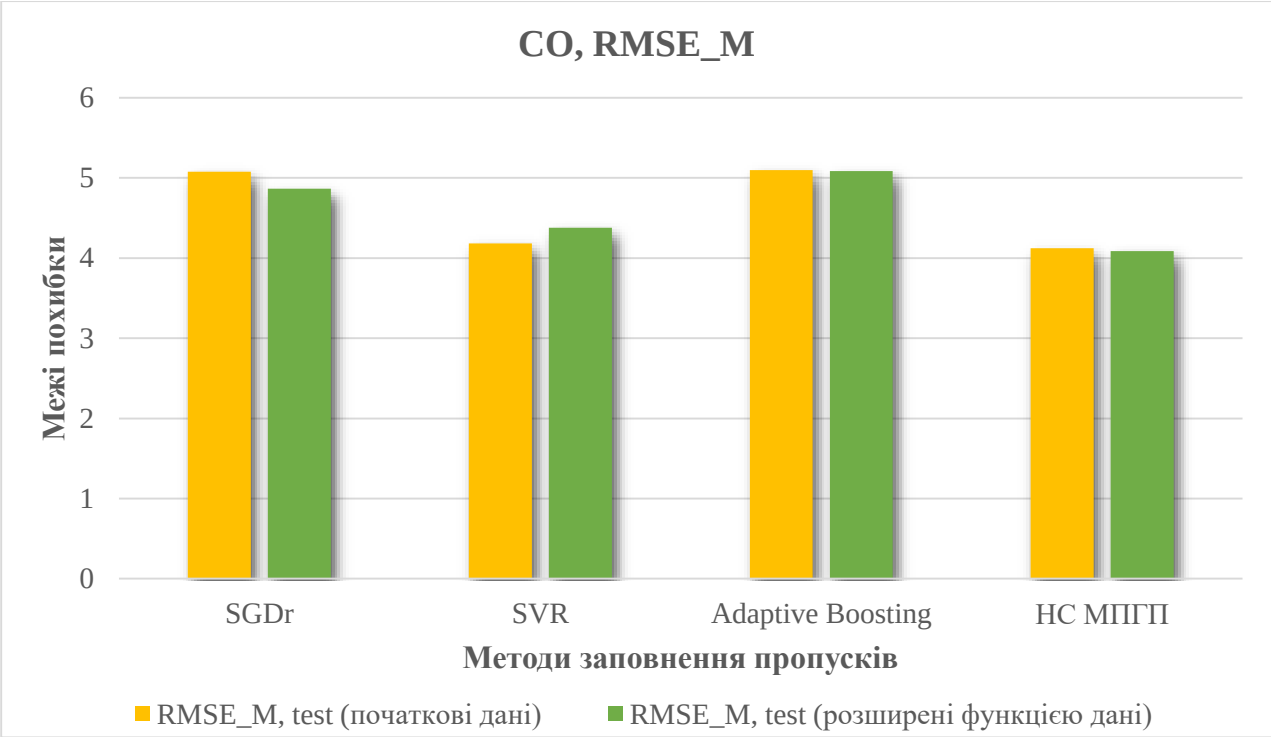


Рис. 4.16б. Тестові похибки заповнення пропусків вуглекислого газу, RMSE_M

Додатково досліджено заповнення пропущених параметрів забруднення атмосферного повітря методом середнього значення на початкових та розширених даних (рис. 4.16в. та 4.16г).

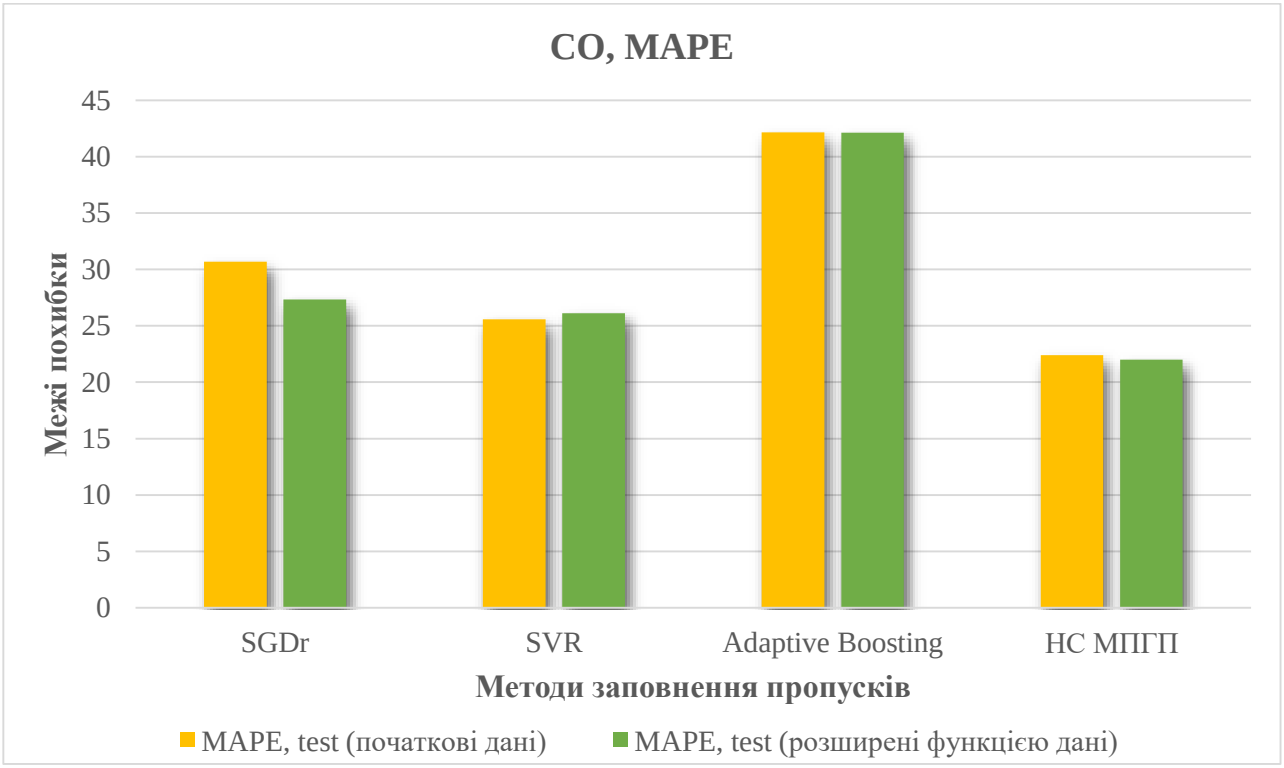


Рис. 4.16в. Тестові похибки заповнення пропусків вуглекислого газу, MAPE

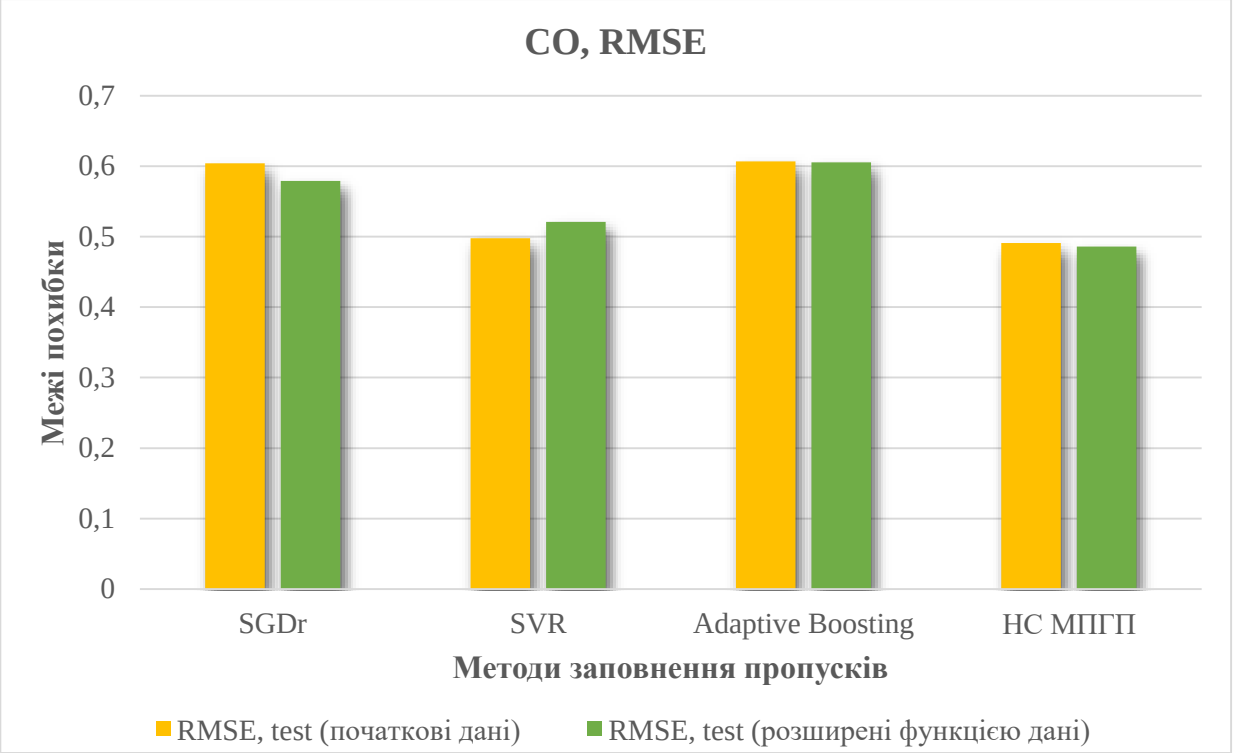


Рис. 4.16г. Тестові похибки заповнення пропусків вуглекислого газу, RMSE

На прикладі оксиду карбону, з таблиці 4.1. та рисунків 4.16а. – 4.16г. випливає, що під час використання удосконаленого методу заповнення пропусків має менші похибки, ніж використання початкових не розширених даних.

Доведено, що похибки MAPE під час застосування нейроподібної структури моделі послідовних геометричних перетворень для заповнення пропущених концентрацій на розширених даних становлять 22,00 % для вуглекислого газу та 20,19 % для діоксиду азоту. Тому, середня похибка застосування нейроподібної структури моделі послідовних геометричних перетворень для заповнення пропущених параметрів забруднення повітряного середовища зменшилася відносно початкової похибки на 4,3 %.

Отже, удосконалення методу нелінійного розширення входів Йох-Хан Пао шляхом використання раціональних дробів є ефективним та може бути використано під час заповнення пропущених концентрацій ПЗб АП. Але прогнозування ПЗб АП на мобільних та вбудованих пристроях вимагає зменшення затрат пам’яті, тому наступним кроком виконано дослідження апроксимації вхідних векторів даних за допомогою їх розширення кластерами.

4.3.1.2.Результати заповнення пропусків за допомогою розробленого методу формування додаткових атрибутів вхідних векторів даних

У розділі також реалізовано та досліджено розроблений метод формування додаткових атрибутів вхідних векторів даних та виконано порівняння методів заповнення пропущених концентрацій ПЗб АП на початкових та розширених даних. Результати досліджень наведено у таблиці 4.2.

Таблиця 4.2.

Похибки заповнення пропущених концентрацій діоксиду азоту

	Похибки	Дані	SGDr	SVR	Adaptive Boosting	НС МПГП
NO_2	**MAE, test**	початкові	21,90005002	18,04394409	19,21834643	17,98804174
		розширені	17,34658	18,68554	18,83271	15,45352
	MAPE, test	початкові	26,62655368	22,22201092	24,55731486	21,40548137
		розширені	20,94597	22,98158	24,13871	18,43131
	RMSE, test	початкові	28,65130623	24,67745703	24,76727793	24,03325785
		розширені	23,23046	24,53463	24,0466	21,2908
	RMSE_M, test	початкові	9,239376404	7,812949704	7,986868084	7,750163769
		розширені	7,491281	8,241156	7,754465	6,865784

Частина результатів застосування досліджених методів заповнення пропущених концентрацій діоксиду азоту графічно наведена на рис. 4.17а. - 4.17г.

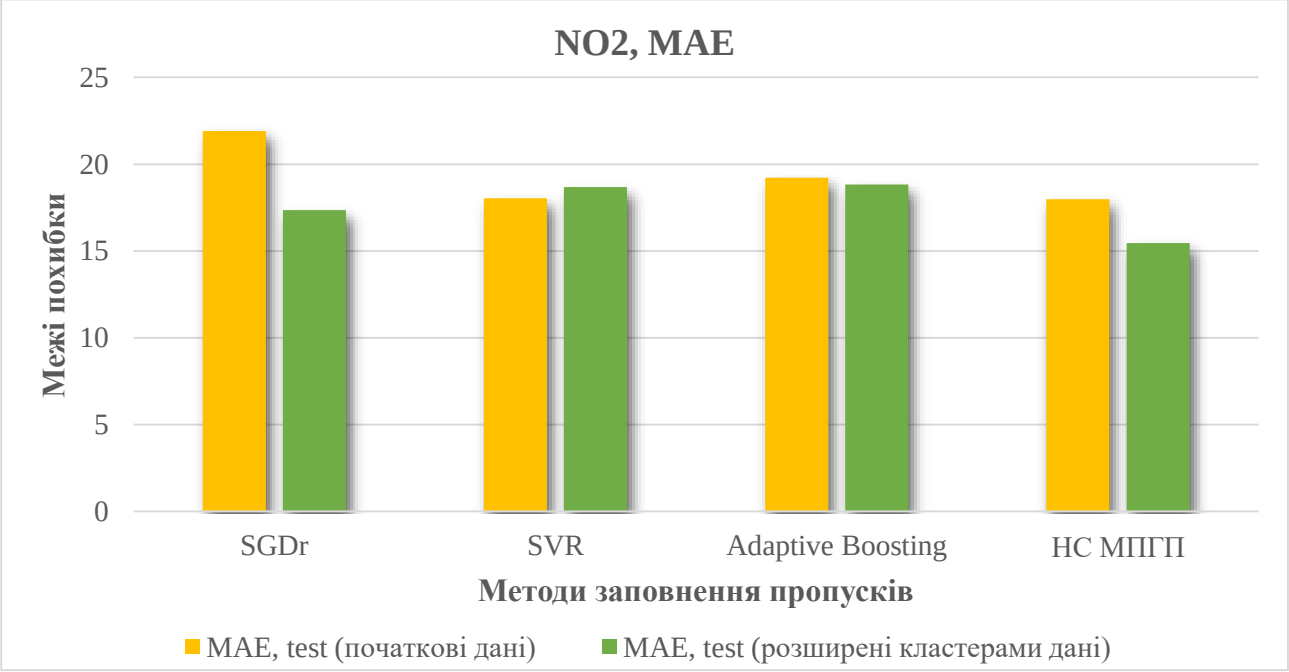


Рис. 4.17а. Тестові похибки заповнення пропусків діоксиду азоту, MAE

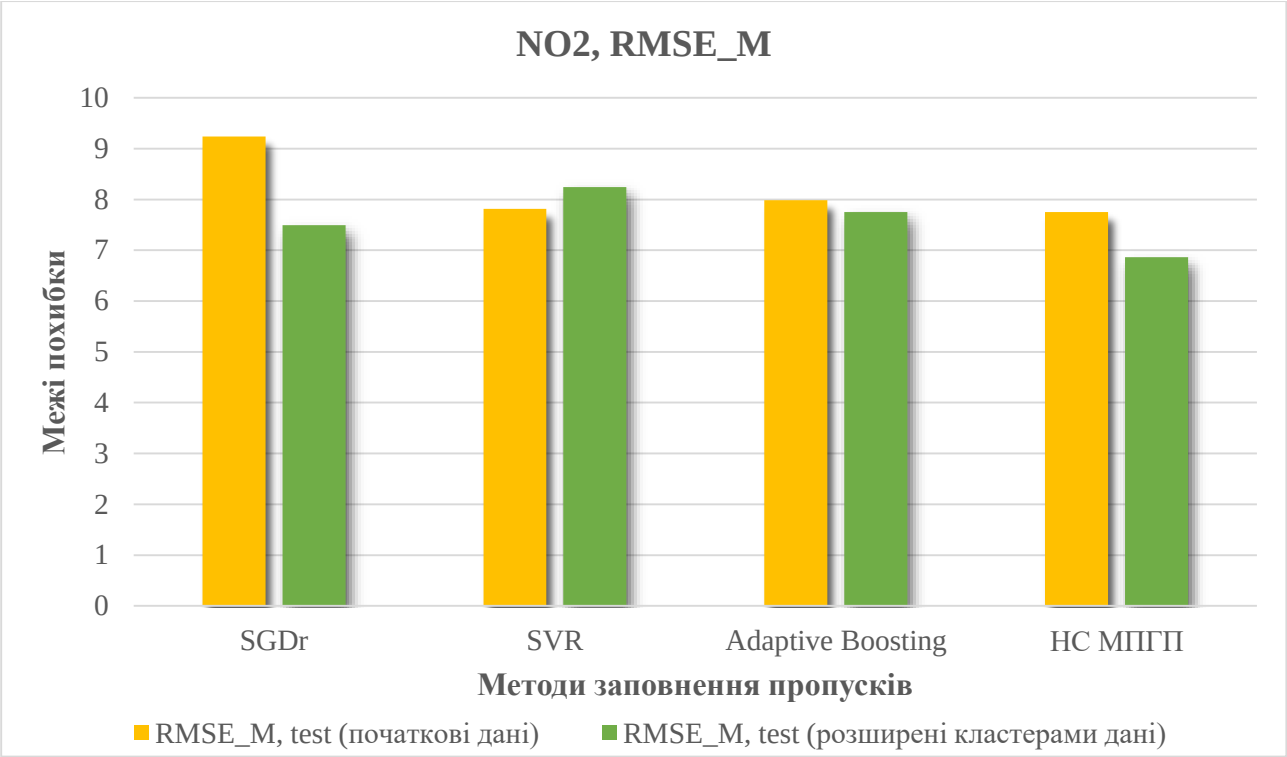


Рис. 4.17б. Тестові похибки заповнення пропусків діоксиду азоту, RMSE_M

Також, виконано дослідження заповнення пропущених концентрацій діоксиду азоту за допомогою моделі на основі усереднення (рис. 4.17в. та 4.17г).

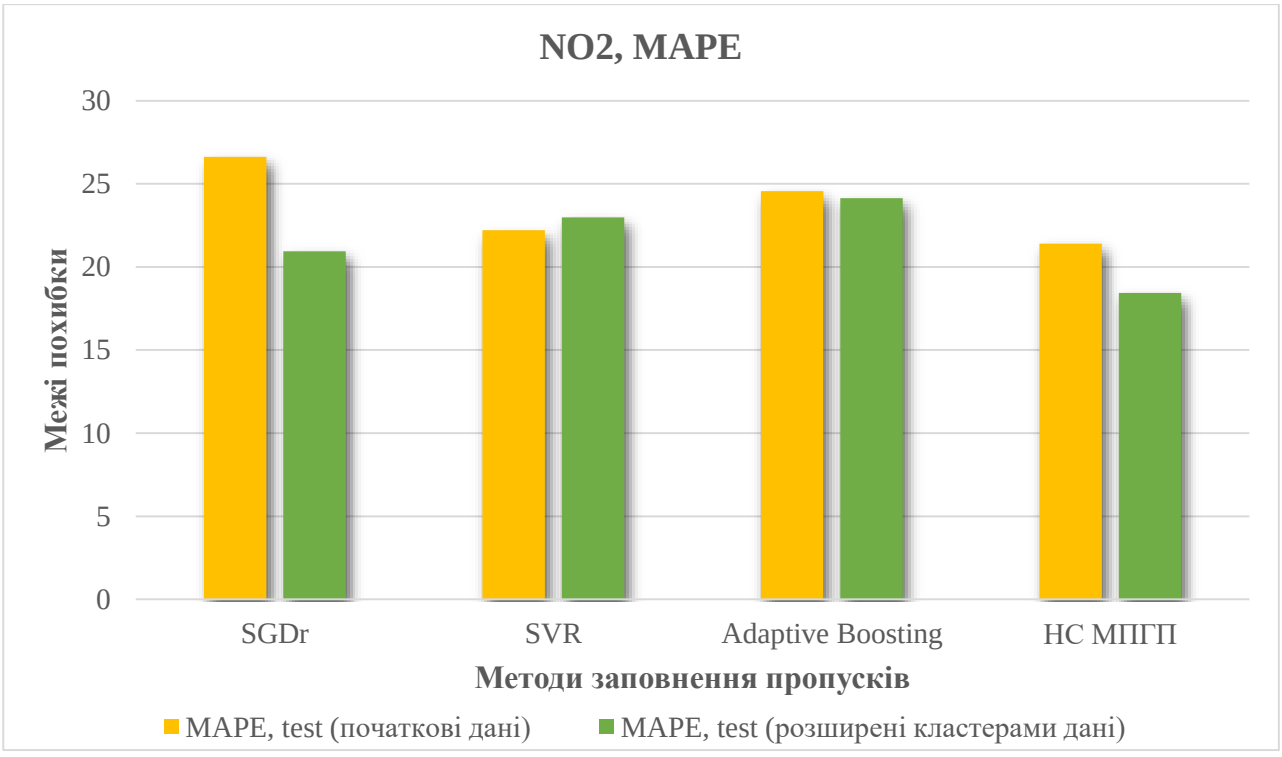


Рис. 4.17в. Тестові похибки заповнення пропусків діоксиду азоту, MAPE

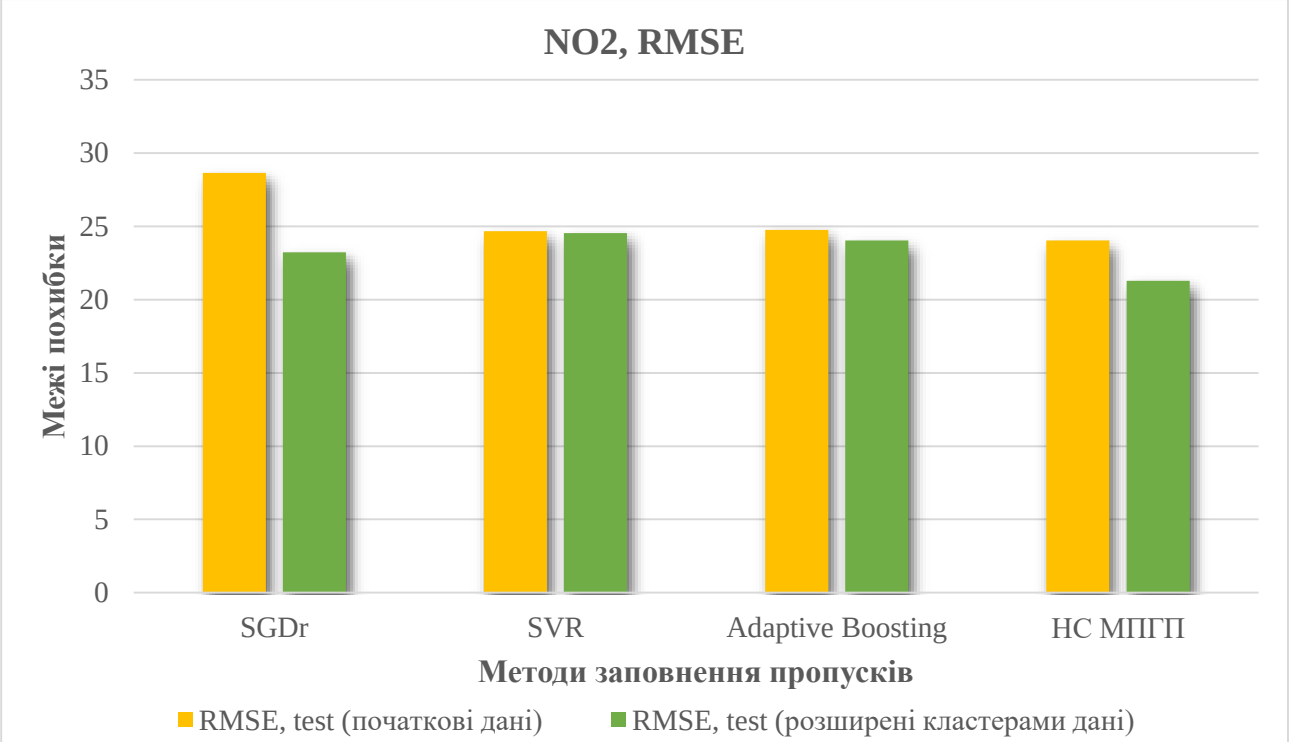


Рис. 4.17г. Тестові похибки заповнення пропусків діоксиду азоту, RMSE

Таким чином, з таблиці 4.1. та рисунків 4.16а. – 4.16г. випливає, що розрболений метод заповнення пропусків має менші похибки, ніж використання початкових не розширених даних, а також ще менші, ніж удосконадений метод.

Досліджено, що похибки MAPE під час застосування нейроподібної структури моделі послідовних геометричних перетворень для заповнення пропущених концентрацій параметрів забруднення атмосферного повітря на розширених даних становлять 21,83 % для вуглекислого газу та 18,43 % для діоксиду азоту. Тому, середня похибка застосування нейроподібної структури моделі послідовних геометричних перетворень для заповнення пропущених параметрів забруднення повітряного середовища зменшилася відносно початкової похибки на 8,5 %.

Тому використання розробленого методу уведення додаткових атрибутів – маркерів кластерів у вектори входів є ефективним методом під час заповнення пропущених концентрацій параметрів забруднення атмосферного повітря, оскільки це забезпечило підвищення точності заповнення пропущених показників параметрів забруднення атмосферного повітря.

4.3.2. Реалізація та оцінка методів прогнозування параметрів забруднення повітряного середовища на мобільних пристроях

Для моделей та методів прогнозування ПЗб АП, в тому числі в умовах пропусків у даних, на етапі навчання виконується пошук найкращих параметрів виконано класом sklearn.model_ selection.GridSearchCV. У цьому класі є метод, котрий на вхід отримує модель та списки для кожного параметра моделі, тоді виконує різні комбінації параметрів.

4.3.2.1.Результати короткотермінового прогнозування концентрацій оксиду карбону шляхом корекції похибки

Пошук моделей та методів виконання точнішого прогнозування ПЗб АП включає в себе попередню підготовку даних. Після розподілу відліків досліджуваного набору викидів концентрацій параметрів забруднення атмосферного повітря ковзними часовими вікнами, виконується згладження вихідного сигналу. Оскільки прогнозується згладжене значення вихідного сигналу, в якому в навчальній вибірці усувається остання, як правило шумова складова, точність передбачення вдається помітно підвищити. В результаті виникає можливість окремо зменшувати похибки різних знаків і помітно підвищити точність прогнозів.

Ефективність розроблюваного методу короткотермінового прогнозування експериментально підтверджено на даних моніторингу забруднення атмосферного повітря, взятих із стаціонарного посту спостереження у місті Києві, що описані раніше. Після розрахунку середньої відносної похибки прогнозування (MAPE) на основі застосування розробленого методу прогнозування доцільним є виконання оцінки його ефективності порівняно з іншими методами. Результати порівняння методу короткотермінового прогнозування концентрацій ПЗб АП на основі корекції похибки за рахунок використання комітет НС різних типів наведені в таблиці 4.3.

Таблиця 4.3.

Похибки прогнозування показників чадного газу за допомогою НС

Похибка прогнозування	Метод 1 (лінійна НС)	Метод 2 (нелінійна НС + RBF)	Метод 3 (комітет НС)
MAPE, %	3,960715 %	3,671264 %	3, 317151 %

Результати прогнозування параметрів забруднення АП досліджуваними методами графічно зображено на рисунку 4.18.

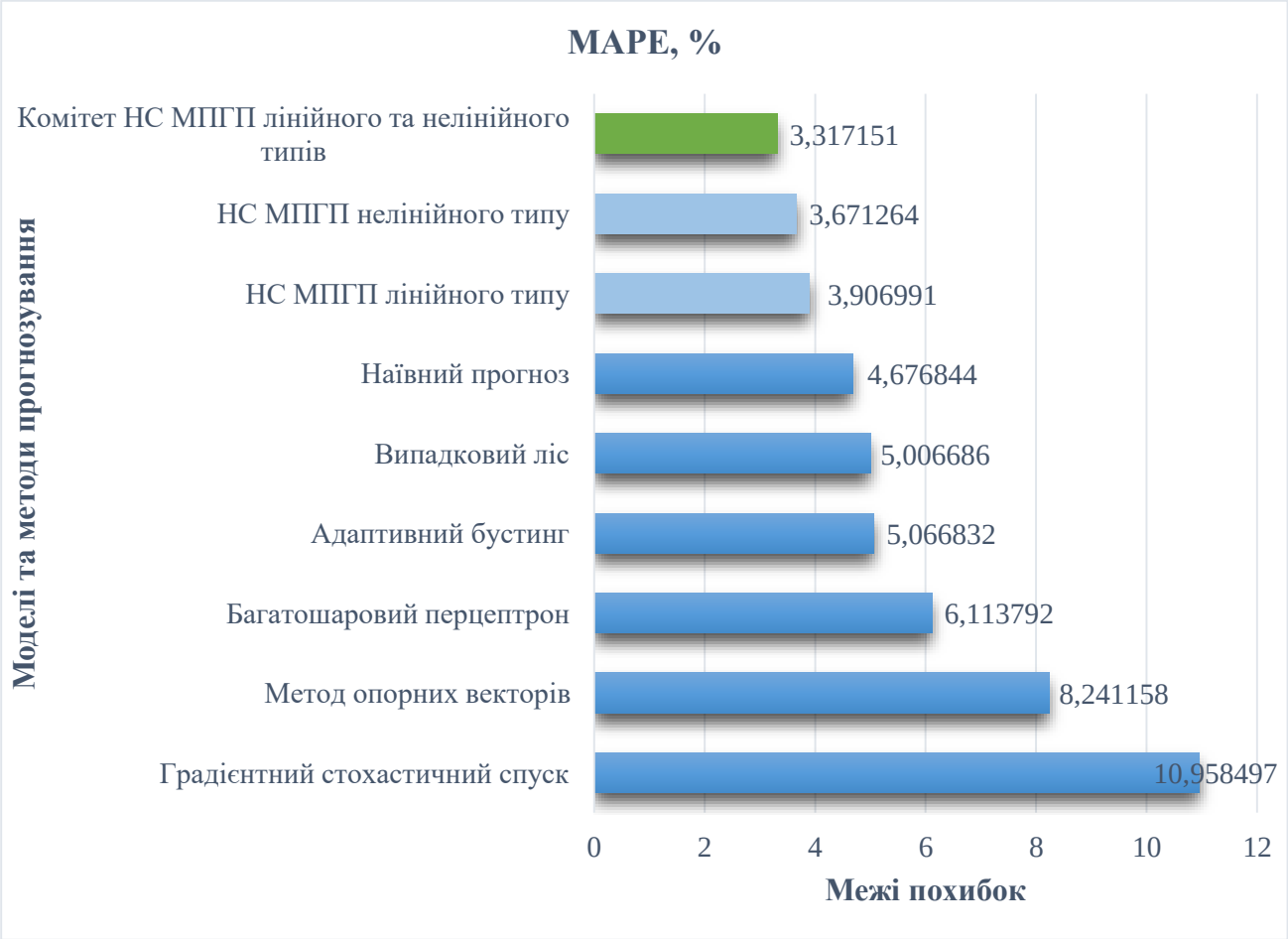


Рисунок 4.18. Середні відносні похибки прогнозування ПЗб АП

З рисунку 4.18 випливає, що розроблений метод короткотермінового прогнозування ПЗб АП за допомогою комітету лінійної та нелінійної нейроподібних структур для часткового коректування окремо додатних і від'ємних відхилень від точних значень зменшив середню відносну похибку застосування на 15 % та забезпечив збільшення горизонту прогнозування на два дні.

4.3.2.2. Результати затрат оперативної пам’яті під час прогнозування концентрацій діоксиду азоту

Використання лінійних поліномів зменшило час прогнозування концентрацій параметрів забруднення атмосферного повітря у 5 разів порівняно із застосуванням НС МПГП та від 2 до 10 разів порівняно з іншими досліджуваними моделями та методами (рис. 4.19.).

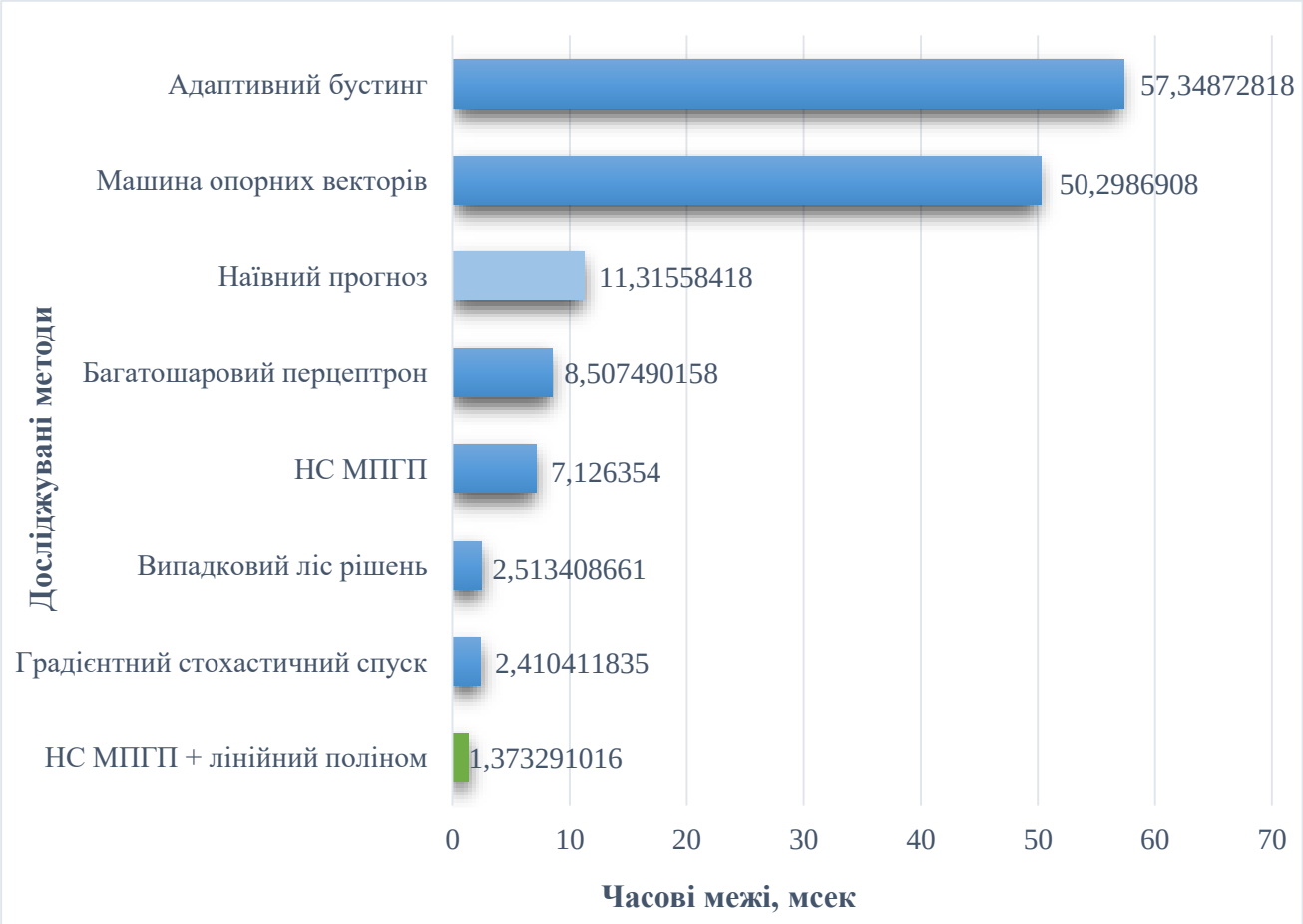


Рис. 4.19. Час застосування досліджених моделей та методів прогнозування параметрів забруднення атмосферного повітря

На рисунку 4.19 наведено, що поєднання нейроподібної структури моделі послідовних геометричних перетворень для навчання та лінійного полінома для прогнозування показує найменший час застосування серед інших досліджуваних методів та моделей. Разом з тим необхідним є зменшення затрат оперативної пам’яті для виконання прогнозування на мобільних та вбудованих пристроях.

Тому кількість затрат оперативної пам’яті обраховуємо на досліджуваних даних моніторингу забруднення атмосферного повітря, де вхідні вектори по n вхідних атрибутах подаються на n входів НС МПГП з одним виходом (рис 3.4.).

Для програмної реалізації НС МПГП з одним виходом в режимі застосування необхідно зберігати в пам’яті $N_{НС\ МПГП}$ значень, що визначаються за жопомогою наступної формули (4.7):

$$N_{НС\ МПГП} = n^2 + 3n - 1, \tag{4.7}$$

де n^2 – синаптичні ваги; $(2n - 1)$ – вхідні атрибути, включаючи входи розширення; та n – вектор початкового зміщення.

У випадку реалізації еквівалентного по функціях лінійного полінома в пам’яті необхідно зберігати $N_{поліном} = n + 1$ значень коефіцієнтів полінома. Використовуючи досліжувані дані з 11 вхідними значеннями визначаємо кількість K разів зменшення затрат оперативної пам’яті за формулою (4.8):

$$K = N_{НС\ МПГП} / N_{поліном} \tag{4.8}$$

Розраховуємо:

$$K = (n^2 + 3n - 1) / (n + 1) = 153 / 12 = 12,75\ (разів).$$

Таким чином, використання нейроподібної структури моделі послідовних геометричних перетворень в поєднання з лінійним поліномом для прогнозування затрачає менше оперативної пам’яті у 12,75 разів менше, ніж використання лише нейроподібної структури моделі послідовних геометричних перетворень.

Отже, доведено, що розроблений метод прогнозування параметрів забруднення атмосферного повітря, котрий базується на застосуванні лінійних нейроподібних структур моделі послідовних геометричних перетворень і побудови на їх основі матриці коефіцієнтів лінійних поліномів є ефективнішим в плані затрат оперативної пам’яті та часу застосування ніж решта досліджених моделей та методів прогнозування.

ВИСНОВКИ ДО РОЗДІЛУ 4

1. Розроблено програмний засіб, де реалізовано розроблені, удосконалені та розвинуті методи за домопогою двох об’єктно-орієнтованих мовах програмування Python та Java. На мові Python виконано створення моделей прогнозування та заповнення пропущених концентрацій параметрів забруднення атмосферного повітря. У Java-фреймворку реалізовано розробку користувацького інтерфейсу та внутрішньої логіки мобільної аплікації.

2. Доведено, що розроблений метод формування додаткових атрибутів вхідних векторів даних забезпечив підвищення точності заповнення пропущених концентрацій параметрів забруднення атмосферного повітря на 8,5 % для НС МПГП. Також досліджено, що розроблений метод формування додаткових атрибутів вхідних веткорів даних спрацьовує не для всіх проаналізованих методів. На прикладі машини опорних векторів точність заповнення пропусків погіршилась на 1,5 %. Досліджено удосконалений нелінійний метод розширення входів Йох-Хан Пао та встановлено, що похибка заповнення пропусків зменшилась на 4,3 %.

3. Виконано дослідження розробленого методу короткотермінового прогнозування параметрів забруднення атмосферного повітря за допомогою комітету лінійної та нелінійної НС для часткового коректування окремо додатних і від’ємних відхилень від точних значень. Встановлено, що похибка прогнозування зменшилась на 15 %, а горизонт прогнозування збільшився на два дні.

4. Досліджено розвинутий метод побудови матриці коефіцієнтів лінійних поліномів, шляхом їх ідентифікації за результатами навчання лінійної НС МПГП для використання на мобільних пристроях та мікроконтролерах. Доведено, що прогнозування параметрів забруднення атмосферного повітря пришвидшилось у 5 разів, а затрати пам’яті зменшились в 12,75 разів.

ВИСНОВКИ

У роботі на основі виконаних теоретичних і експериментальних досліджень розв’язано актуальне наукове завдання, - розроблено нейроподібні методи та засоби з неітеративним навчанням та підвищеною точністю прогнозування параметрів забруднення атмосферного повітря, в тому числі в умовах частково пропущених параметрів у даних моніторингу повітряного середовища.

Під час вирішення завдань дисертаційної роботи отримано наступні основні результати:

1. Проаналізовано актуальні задачі моніторингу навколишнього середовища та особливості існуючих методів і засобів для здійснення функцій прогнозування параметрів забруднення атмосферного повітря, зокрема в умовах пропусків у даних моніторингу повітряного середовища.
2. Вперше розроблено метод коректування похибки короткотермінового прогнозування параметрів забруднення повітряного середовища в реальному часі на основі комітету нейроподібних структур різних типів, що забезпечило зменшення похибки прогнозування на 15% та збільшило горизонт прогнозування на два дні.
3. Вперше розроблено метод введення додаткових ознак за допомогою попередньої кластеризації вхідних векторів даних, що підвищило вірогідність відновлення пропущених компонент параметрів забруднення атмосферного повітря на 2,5-14,5% в залежності від виду параметру. Встановлено, що під час використання розробленого методу для завдання заповнення пропусків у даних моніторингу атмосферного повітря на основі машини опорних векторів зменшує точність заповнення пропущених компонент на тестових даних на 1-2% залежно від параметру забруднення повітряного середовища.

4. Розвинуто метод побудови апроксимаційних поліномів шляхом ідентифікації їх коефіцієнтів за результатами навчання відповідних нейроподібних структур, що забезпечило зменшення затрат пам’яті під час прогнозування параметрів забруднення атмосферного повітря в 12,75 разів.
5. Удосконалено метод функційного розширення входів Йох-Хан Пао шляхом застосування раціональних дробів, що забезпечило зниження похибок заповнення пропущених компонент параметрів забруднення атмосферного повітря за допомогою нейроподібної структури моделі послідовних геометричних перетворень в режимі використання на невідомих при навчанні даних на 2,6-6% в залежності від виду параметру.
6. Розроблено програмний засіб з набором бібліотек реалізацій методів прогнозування параметрів забруднення повітряного середовища, зокрема в умовах пропусків у даних моніторингу атмосферного повітря.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Джигирей В. С. Основи екології та охорона навколишнього природного середовища / В. С. Джигирей, В. М. Сторожук, Р. А. Яцюк // Екологія та охорона природи. — Львів: Афіша, 2000. — 272 с.
2. Air quality – General aspects – Vocabulary : ISO 4225:1994(en). — [Reviewed and confirmed in 2015]. — 12 p.
3. Chaudhryl V. Arduair: Air Quality Monitoring / V. Chaudhryl // International Journal of Environmental Engineering and Management. — 2013. — P. 639–646.
4. Zhanga H. Air pollution and control action in Beijing / H. Zhanga, Sh. Wangb, J. Haob, X. Wangc, Sh. Wanga, F. Chaia, M. Lid // Journal of Cleaner Production. — Volume 112. — Part 2. — January 2016. — P. 1519–1527.
5. Боголюбов В. М. Моніторинг довкілля : підручник / [Боголюбов В. М., Клименко М. О., Мокін В. Б. та ін.] ; за ред. В.М. Боголюбова. і Т.А. Сафранова. — Херсон : Грінь Д.С., 2011. — 530 с.
6. Gómez-Losada A. Characterization of background air pollution exposure in urban environments using a metric based on Hidden Markov Models / A. Gómez-Losadaa, J. C. M. Piresb, R. Pino-Mejíasa // Atmospheric Environment. — 2016. — Volume 127. — P. 255–261.
7. Ничик О.В. Моніторинг довкілля: Курс лекцій для студентів напряму 6.040106 "Екологія, охорона навколишнього середовища та збалансоване природокористування" / Ничик О. В. — К.: НУХТ, 2011. — 67 с.
8. Mishchuk O. Missing Data Imputation Through SGTM Neural-Like Structure for Environmental Monitoring Tasks / O. Mishchuk, R. Tkachenko, I. Izonin // Advances in Computer Science for Engineering and Education II. — International Conference on Computer Science, Engineering and Education Applications, ICCSEEA 2019. — Springer. — Vol. 938. — P. 142–151.
9. Доронина Ю. В. Реинжиниринг мониторинговых информационных систем циклического типа / Ю. В. Доронина // Східно-Європейський журнал передових технологій. — Харків, 2012. — № 1/2(55). — С. 12–14.

10. Дзюба С. М. Применение информационных технологий для решения задач экологического мониторинга загрязнения атмосферы мегаполисов / С. М. Дзюба, Н. В. Белянина, М. Н. Прокопенко, С. А. Серовиков // Вісник Національного технічного університету «ХПІ». Тематичний випуск: Інформатика і моделювання. — 2010. — № 21. — С. 58–65.
11. Шипулін В. Д. Основні принципи геоінформаційних систем / В. Д. Шипулін. — Харків : ХНАМГ, 2012. — 312 с.
12. Скриник О. А. Відновлення пропусків у часових рядах метеорологічних показників / О. А. Скриник, О. Я. Скриник // Наукові праці УкрНДГМІ. — 2011. — Вип. 260. — С. 46–53.
13. Рогожин О. Г. Інформаційний інструментарій оцінки екологічних ресурсів в Україні / О. Г. Рогожин, Є. В. Хлобистов, Є. О. Яковлев // Математичне моделювання в економіці — К., 2015. — No3. — С. 13–26.
14. Гладкий А. В. Методи числового моніторингу екологічних процесів: Навчальний посібник / А. В. Гладкий, В. В. Скопець. — К.: ІВЦ «Видаництво «Політехніка», ТОВ «Фірма «Періодика», 2005. — 152 с.
15. Munn R. E. Policy Making in an Era of Global Environmental Change / R. E. Munn. – Springer. — 1996. — 224 p.
16. Снытко В. А., Собисевич А. В. Система экологического мониторинга в научном наследии академиков И. П. Герасимова и Ю. А. Израэля / А. В. Снытко, А. В. Собисевич // Труды пятой международной научно-практической конференции «Индикация состояния окружающей среды: теория, практика, образование», 30 ноября - 3 декабря 2017 года: сборник статей. — 2017. — С. 393–398.
17. Снытко В. А., Собисевич А. В. Вклад академика И. П. Герасимова в развитие основ мониторинга природной среды / А. В. Снытко, А. В. Собисевич // Мониторинг состояния и загрязнения окружающей среды. — Основные результаты и пути развития тезисы докладов Всероссийской научной конференции. — ФГБУ «Институт глобального климата и экологии Росгидромета и РАН». — М., 2017. — С. 24–26.

18. Баньковский Л. В. Опасные ситуации природного характера: Учебно-методическое пособие: Ч. 1. — 2-е изд.. — Соликамск: РИО ГОУ ВПО «СГПИ», 2008. — 230 с.
19. Погребенник В. Екологічний моніторинг: концепції, принципи, системи / В. Погребенник, М. Мельник, М. Бойчук // Вимірювальна техніка та метрологія. — № 65. — 2005. — С. 164–171.
20. Голубець М. А. Екологічний потенціал наземних екосистем / М. А. Голубець. — Львів : Поллі, 2003. — 180 с.
21. Shakovska N. The structure of Information Systems for Environmental Monitoring / N. Shakovska, O. Shamuratov // XI-th International Scientific and Technical Conference «Computer Science and Information Technologies. » — September 6-10. — Lviv, Ukraine. — 2016. — pp. 102–107.
22. Данилко В. К. Інформаційні ресурси стану забруднення й охорони атмосферного повітря та їх аналіз / В. К. Данилко, О. Ю. Борецька // Вісник ЖДТУ №3(41): Економічні науки. — 2007. — С. 145–154.
23. Северин Л. І. Природоохоронні технології. Частина 1. Захист атмосфери / Л. І. Северин, В. Г. Петрук, І. І. Безвозюк, І. В. Васильківський. — Вінниця: ВНТУ, 2012. — 388 с.
24. Мислива Т. М. Основи моніторингу довкілля: навч. посібник / Т. М. Мислива, М. Й. Долгілевич. — Житомир: Вид-во ДВНЗ «Державний агроекологічний університет», 2007. — 376 с.
25. Revathy V. S. Air Pollution Monitoring System / V. S. Revathy, K. Ganesan, K. Rohini, S. Tamil Chindhu, T. Boobalan // Journal of Electronics and Communication Engineering. — 2016. — Volume 11. — Issue 2. — Ver. II. — P. 27–40.
26. Satish, U. Is CO2 an Indoor Pollutant? Direct Effects of Low-to-Moderate CO2 Concentrations on Human Decision-Making Performance / U. Satish, M. J. Mendell, K. Shekhar, T. Hotchi and other // Environ. Health Perspect. — 2012. — Volume 120. — P. 1671–1677.

27. Persily A. Carbon dioxide generation rates for building occupants / A. Persily, L. De Jonge // Indoor Air. — 2017. — Volume 27. — P. 868–879.
28. Jianhui B. Study on surface O_3 chemistry and photochemistry by UV energy conservation / B. Jianhui, N. Chauhan, S. Singh, T. Saud, M. Saxena and other // Atmospheric Pollution Results. — 2010. — P. 118–127.
29. Sarbu I. Ecological refrigerants used in refrigeration, air-conditioning and heat pump systems / I. Sarbu, E. S. Valea // Poceedings of International Conference on Power Systems, Energy, Environment. — Interlaken, Switzerland. — 2014. — P. 178–184.
30. Сніжко С. І. Оцінка сучасного рівня забруднення атмосферного повітря у м. Києві / С. І. Сніжко, О. Г. Шевченко, Д. П. Скляренко // Вісник Київського національного університету імені Тараса Шевченка. Географія. — 2005. — № 51. — С. 28–30.
31. Salnikov V. G. The Impact of Air Pollution on Human Health: Focusing on the Rudnyi Altay Industrial Area / V. G. Salnikov, M. A. Karatayev // American Journal of Environmental Sciences. — Volume 7 (3). — 2011. — P. 286–294.
32. Darbe Ph. D. Overview of air pollution and endocrine disorders / Ph. D. Darbre // International Journal of General Medicine. — Vol. 11. — 2018. — P. 191–207.
33. Регіональна доповідь про стан навколишнього природного середовища Київської області у 2015 році [Текст]. — Київ: Київська ОДА. — 2016. — 233 с.
34. Кольцов М. Моніторинг якості атмосферного повітря: український та міжнародний досвід. [Аналітична записка] / М. Кольцов, Л. Шевченко. — Київ: ГО «Фундація «Відкрите Суспільство», 2018. — 13 с.
35. Регіональна доповідь про стан навколишнього природного середовища Київської області у 2018 році [Текст]. — Київ: ОДА. — 2019. — 256 с.
36. Harmens H. Air Pollution: Deposition to and impacts on vegetation in (South)-East Europe, Caucasus, Central Asia (EECCA/SEE) and South-East Asia. [Report] / H. Harmens, G. Mills G. — ICP Vegetation Programme Coordination Centre, Centre for Ecology and Hydrology. — Bangor. UK. — 2014. — 72 p.

37. Степико М. Т., Барков Ю. Ю. Стратегічний прогноз як об'єкт дослідження // Стратегії розвитку України: теорія і практика / За ред. О. С. Власюка. — К.: НІСД, 2002. — С. 38–50.
38. Про охорону атмосферного повітря: Закон України від 16.10.1992 №2707-XII (у чинній редакції від 18.12.2017). Відомості Верховної Ради України (ВВР), 1992, № 50, ст.678.
39. Mendez D. P-Sense: A participatory sensing system for air pollution monitoring and control / D. Mendez, A. J. Perez, M. A. Labrador, J. J. Marron // Pervasive Computing and Communications Workshops. — 2011. — P. 344–347.
40. RD 52.04.186-89 «Guidelines for controlling atmospheric pollution». Accessed on: August 10, 2019. [Online]. Available: http://docs.cntd.ru/document/1200036406
41. Nickovic S. A model for prediction of desert dust cycle in the atmosphere. / Nickovic S., Kallos G., Papadopoulos A., Kakaliagou O. // J Geoph Res. — No. 106. — 2001. — P. 18113–18129.
42. Brauer, M., et al.: Exposure assessment for estimation of the global burden of disease attributable to outdoor air pollution. Environmental Science & Technology. — Vol. 42, No. 2. — 2012. — P. 652–660. DOI: 10.1021/es2025752
43. Kuznietsova N. V.Business Intelligence Techniques For Missing Data Imputation / N. V. Kuznietsova, P. I. Bidyuk // Research bulletin of NTUU “KPI”. — 2015. — No. 5. — P. 47–56.
44. Кузнєцова Н. В. Виявлення та оброблення невизначеностей у формі неповних даних методами інтелектуального аналізу // Системні дослідження та інформаційні технології. — 2016. — № 2. — С. 104–115.
45. Злоба Е. Статистические методы восстановления пропущенных данных / Е. Злоба, И. Яцкив // Computer Modelling & New Technologies. — 2002. — Вип. №6 (1). — С. 51–61.
46. Зангиева И. К. Сравнение эффективности алгоритмов заполнения пропусков в данных в зависимости от используемого метода анализа /

И. К. Зангиева, Е. С. Тимонина // Компьютерные и информационные науки». — № 1 (119). — 2014. — С. 41–55.

47. Мокін В. Б. Інформаційні технології автоматизації обробки параметрів геоінформаційних систем з геометричними мережами : монографія / В. Б. Мокін, В. Г. Сторчак, Є. М. Крижановський, О. В. Гавенко, В. Ю. Балачук. — Вінниця : ВНТУ. — 2014. — 196 с.
48. Мокін В. Б. Розробка підсистеми реєстрації та попередньої обробки даних контролю шкідливих викидів / В. Б. Мокін, Г. В. Горячев, Д. І. Катєльніков, С. О. Жуков, І. А. Моргун // Вісник Вінницького політехнічного інституту. Спеціальний випуск за матеріалами І-го Всеукраїнського з'їзду екологів. — 2006. — №5 — С. 124–128.
49. Newman D. Missing Data: Five Practical Guidelines // Organizational Research Methods. — 2014. — №17(4). — P. 372–411.
50. Graham, J. W. Multiple imputation in multivariate research / J. W. Greham, S. M. Hofer // Modeling longitudinal and multiple-group data: Practical issues, applied approaches, and specific examples. — Hillsdale, NJ: Erlbaum. — 2000. — P. 201–218.
51. Schafer J. L. Missing data: Our view of the state of the art / J. L. Schafer, J. W. Graham // Psychological Methods 7. — 2002. — P. 147−177.
52. Graham J. W. How many imputations are really needed? Some practical clarifications of multiple imputation theory / J.W. Graham, A.E. Olchowski, T.D. Gilreath // Prevention Science. — 2007. — 8. — P. 206–213.
53. Karahalios A. A review of the reporting and handling of missing data in cohort studies with repeated assessment of exposure measures / A. Karahalios, L. Baglietto, J. D. Carlin, D. R. English and J. A. Simpson // BMC Med Res Methodology. — 2012. — 12. — 96 p.
54. Van Buuren S. Flexible Imputation of Missing Data // Chapman and Hall/CRC. — 2012. — 342 p.
55. Скриник О. А. Відновлення пропусків у часових рядах метеорологічних показників / О. А. Скриник, О. Я. Скриник // Наукові праці Українського

науково-дослідного гідрометеорологічного інституту: Зб. наук. пр. — 2011. — Вип. 260. — С. 46–53.

56. Baraldi A. N. An introduction to modern missing data analyses / A. N. Balardi, C. K. Enders // Journal of School Psychology, 2010. — Vol. 48, No. 1. — P. 5–37. DOI: 10.1016/j.jsp.2009.10.001
57. Soley-Bori M. Dealing with missing data: key assumptions and methods for applied analysis // Technical Report. — 2013. — No. 4. — P. 1–20.
58. Luengo J. On the choice of the best imputation methods for missing values considering three groups of classification methods / J. Luengo, S. Garcia, F. Herrera // Knowledge and Information Systems. — 2012. — Volume 32. — P. 77–108.
59. Xiaoping Zhu Comparison of Four Methods for Handing Missing Data in Longitudinal Data Analysis through a Simulation Study // Open Journal of Statistics. — 2014. — No. 04(11). — P. 933-944. DOI: 10.4236/ojs.2014.411088
60. Слабченко О. О. Інформаційна технологія імпутації даних змішаної природи в задачах інтелектуального аналізу / О. О. Слабченко // Проблеми інформаційних технологій. — 2016. — № 01. — С. 155–161.
61. Kaminskyi S. Recovery Gaps in Experimental Data / S. Kaminskyj, N. Kunanets, et al. // Proceedings of the 2nd International Conference on Computational Linguistics and Intelligent Systems. — Volume I: Main Conference Lviv, Ukraine, June 25-27. — 2018. — No. 1. — P. 110–118.
62. Horton N. J. Much ado about nothing: A comparison of missing data methods and software to fit incomplete data regression models / N. J. Horton, K. P. Kleinman // Am. Stat. 2007. — No. 61. — P. 79–90.
63. Silva-Ramirez E.-L. Missing value imputation on missing completely at random data using multilayer perceptrons / E.-L. Silva-Ramirez, R. Pino-Mejías, M. López-Coello, M.-D. Cubiles-de-la-Vega // Neural Networks. — 2011. — Vol. 24, Issue 1. — P. 121–129. DOI: 10.1016/j.neunet.2010.09.008
64. Matloff N. Statistical Regression and Classification: From Linear Models to Machine Learning. — Chapman and Hall/CRC Press, Davis. — 2017. — 528 p.

65. Zhang Z. Missing data imputation: focusing on single imputation / Annual Transl Med. — 2016. — No. 4(1). — 9 p.
66. Снитюк В.Е. Эволюционный метод восстановления пропусков в данных // Сборник трудов VI-й межд. конф. «Интеллектуальный анализ информации». — Киев. — 2006. — С. 262-271.
67. Enders C. K. Applied missing data analysis / New York. — NY: Guilford Press. — 2010. — 377 p.
68. Zhou X.-Y., Lim J. S. EM algorithm with GMM and naive bayesian to implement missing values // Advanced Science and Technology Letters. — 2014. — Vol. 46. — P. 1–5.
69. Barlett J. W. Multiple imputation of covariates by fully conditional specification: Accommodating the substantive model / J. W. Bartlett, I. R. White, Sh. R. Seaman, J. R. Carpenter // Stat Methods Med Res. — 2015. — No. 24(4). — P. 462-87. DOI: 10.1177/0962280214521348
70. Estrabrook R. A Comparison of Factor Score Estimation Methods in the Presence of Missing Data: Reliability and an Application to Nicotine Dependence / R. Estabrook, M. Neale // Multivariate Behav Res. — 2013. — No. 48(1). — P. 1–27. DOI: 10.1080/00273171.2012.730072
71. Tian J. Missing data analyses: a hybrid multiple imputation algorithm using Gray System Theory and entropy based on clustering / J. Tian, B. Yu, D. Yu, et al. // Applied Intelligance. — 2014. — No. 40. — P. 376–388. DOI: 10.1007/s10489-013-0469-x
72. Zhu B. A robust missing value imputation method for noisy data / B. Zhu, C. He, P. Liatsis // Applied Intelligance. — 2012. — No. 36. — P. 61–74. DOI: 10.1007/s10489-010-0244-1
73. Dong Y. Principled missing data methods for researchers / Y. Dong, CY. J. Peng // SpringerPlus. — 2013. — No. 2 (222). DOI: 10.1186/2193-1801-2-222

74. Martyshenko S. N. Methods for recovering omissions in data presented in various measuring scales // Territory of new opportunities. — 2013. — No. 4. — P. 242— 255.
75. Alonso A. M. Resampling time series using missing values techniques / A. M. Alonso, D. Peña, J. Romo // Annuals of the Institute of Statistical Mathematics. — 2003. — No. 55 (4). — P. 765–796. DOI: 10.1007/BF02523392
76. Davison A. Resampling Variance Estimation in Surveys with Missing Data / A. Davison, S. Sardy // Journal of Official Statistics. — 2007. — No. 23 (3). — P. 371–386.
77. Khayati M. Scalable recovery of missing blocks in time series with high and low cross-correlations / M. Khayati, P. Cudré-Mauroux, M. H. Böhlen // Knowl Inf Syst. — 2020. — No. 62. — P. 2257–2280. DOI: 10.1007/s10115-019-01421-7
78. Міщук О.С. Методи оброблення та заповнення пропущених параметрів у даних екологічного моніторингу / О.С. Міщук, Р.О. Ткаченко // Науковий вісник НЛТУ України. — 2019. — №29(6). — С. 119–122. DOI: 10.15421/40290623
79. Chaudhry A. A Method for Improving Imputation and Prediction Accuracy of Highly Seasonal Univariate Data with Large Periods of Missingness / A. Chaudhry, W. Li, et al. // Wireless Communications and Mobile Computing. — 2019. — No. 3. — P. 1–13. DOI: 10.1155/2019/4039758
80. Maheswari K. Missing Data Handling by Mean Imputation Method and Statistical Analysis of Classification Algorithm / K. Maheswari, P. Amutha Priya, S. Ramkumar, M. Arun // Springer Innovations in Communication and Computing. — 2020. — P. 137-149. DOI: 10.1007/978-3-030-19562-5_14
81. Honghai F. A SVM regression based approach to filling in missing values / F. Honghai, C. Guoshun, Y. Cheng, Y. Bingru, C. Yumei // Proceedings of the 9th international conference on Knowledge-Based Intelligent Information and Engineering Systems. — KES 2005. — Springer, Berlin, Heidelberg. — Vol. 3683. — P. 581— 587. DOI: 10.1007/11553939_83

82. Pelckmans K. Handling missing values in support vector machine classifiers / K. Pelckmans, J. D. Brabanter, J. A. K. Suykens, B. D. Moor // Neural Networks. — 2005. — No. 18. — P. 684–692.
83. Zhang T. Solving Large Scale Linear Prediction Problems Using Stochastic Gradient Descent Algorithms // ICML 2004: Proceedings of the twenty-first international conference on machine learning. — Omnipress. — 2004. — P.1–8.
84. Peleshko D. Research of usage of Haar-like features and AdaBoost algorithm in Viola-Jones method of object detection / D. Peleshko, K. Soroka // 12th International Conference on the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM). — Polyana Svalyava. — 2013. — P. 284–286.
85. Murphy K. V. Machine Learning: A Probabilistic Perspective // Massachusetts Institute of Technology, Library of Congress Cataloging-in-Publication Information. — MIT Press, Cambridge, MA. — 2012. — 1067 p.
86. Moore M. A decision tree approach to modeling the private label apparel consumer / M. Moore, J.M. Carpenter // Mark. Intell. Plan. — 2010. — No. 28(1). — P. 59–69.
87. Cobourn G. An enhanced PM2.5 air quality forecast model based on nonlinear regression and back-trajectory concentrations // Atmos. Environ. — 2010. — Volume 44, Issue 25. — P. 3015-3023. DOI: 10.1016/j.atmosenv.2010.05.009
88. Кіптенко Є. М. Прогнозування рівнів високого забруднення атмосферного повітря в містах України / Є. М. Кіптенко, Т. В. Козленко // Тр. УкрНДГМІ. — 2002. — Вип. 250. — С. 288–297.
89. Дзендзелюк О. Автоматизована система моніторингу параметрів довкілля / О. Дзендзелюк, І. Мусійчук, В. Рабик // Теор. електротехніка. — 2010. — Вип. 61. — С. 90–98.
90. Hurley P. Yearlong, high-resolution, urban airshed modeling: verification of TAPM predictions of smog and particles in Melbourne, Australia / Hurley P. Yearlong, Manins P, Lee S., Boyle R., Ng Y.L., Dewundege P. // Atmos Environ. — 2003. — No. 37(1899). — P. 910–916.

91. Лопатко О. Нейронні мережі як засіб прогнозування значення температури за перехідним процесом / О. Лопатко, І. Микитин // Вимірювальна техніка та метрологія. — 2016. — №77. — С. 65–70.
92. Karpinski M. Simulation of artificial neural networks for assessing the ecological state of surface water / Karpinski M., Pohrebennyk V., Bernatska N., Ganczarczyk J., Shevchenko O. // 18th International Multidisciplinary Scientific Geoconference. — Albena, Bulgaria. — 2018. — Vol. 18, Issue 2.1. — P. 693–700. DOI: 10.5593/sgem2018/2.1/S07.088
93. Kriesel D. A Brief Introduction to Neural Networks. — 2007. — [Online]. Available: http://www. dkriesel.com/en/science/neural_networks
94. Nezhad H. B. New neural network-based response surface method for reliability analysis of structures / H.B. Nezhad, M. Miri, and M. Ghasemi // Neural Computing and Applications. — Springer, London. — 2017. — P. 1–15.
95. James G. An introduction to statistical learning / G. James, D. Witten, T. Hastie, R. Tibshirani // Springer Science. — Business Media NY. — 2013. — 426 p.
96. Дзендзелюк О. Прогнозування параметрів довкілля на основі штучних нейронних мереж / О. Дзендзелюк, З. Любунь, В. Рабик // Електроніка та інформаційні технології. — 2015. — Вип. 5. — С. 102–113.
97. Kolmykov V. Comparative analysis of the statistical model and the neural network of reverse distribution in the prediction task // Applied Informatics. — Litrus. — 2010. — No. 6 (30) . — P. 111–118.
98. Schmidhuber J. Exploring the predictable // Advances in Evolutionary Computing, eds A. Ghosh, S. Tsutsui. — B: Springer. — 2003. — P. 579–612.
99. Denil M. Predicting parameters in deep learning / M. Denil, B. Shakibi, L. Dinh, N. de Freitas // Advances in neural information processing systems. — Red Hook, NY: Curran. — 2013. — Vol. 26. — P. 2148–2156.
100. Ganesh S. S. Forecasting air quality index using regression models: A case study on Delhi and Houston / S. S. Ganesh, S. H. Modali, S. R. Palreddy, P. Arulmozhivarman // International Conference on Trends in Electronics and Informatics (ICEI). — 2017. — P. 248–254.

101. Huixiang L. Air quality index and air pollutant concentration prediction based on machine learning algorithms / L. Huixiang, L. Qing, Y. Dongbing, G. Yu. — Vol 9. — No. 4069. — 2019. DOI: 10.3390/app9194069
102. Mekanik F. Multiple regression and Artificial Neural Network for long-term rainfall forecasting using large scale climate modes / F. Mekanik M. Imteaz, S. Gato-Trinidad, A. Elmahdi // J Hydrol. — 2013. — No. 503. — P. 11–21
103. Pelliccioni A. Air pollution model and neural network: an integrated modelling system / A. Pelliccioni, T. Tirabassi // IL NUOVO CIMENTO. — May, 2008. — P. 22–23.
104. Бидюк П. И. Системный подход к построению регрессионной модели по временным рядам / П. И. Бидюк, И. В. Баклан // Системні дослідження та інформаційні технології. — 2002. — № 3. — С. 110–135.
105. Schornobay-Lui E. Prediction of short and medium term PM10 concentration using artificial neural networks / E. Schornobay-Lui, E. Alexandrina, M. Aguiar, W. Hanisch, E. Corrêa, N. Corrêa // Management of Environmental Quality. — 2019. — No. 30(2). — P. 414–436.
106. Gokhale Sh. A hybrid model for predicting carbon monoxide from vehicular exhausts in urban environments / Sh. Gokhale, and M. Khare // Atmospheric Environment. — 2005. — No. 39(22). — P. 4025–4040. DOI: 10. 1016/j.atmosenv.2005.04.010
107. Rahman N. H. A. Artificial neural networks and fuzzy time series forecasting: an application to air quality / N. H. A. Rahman, M. H. Lee, M. T. Latif // Quality and Quantity. — 2014. — No. 49(6). — P. 2633–2647.
108. Husaini N. A. Jordan pi-sigma neural network for temperature prediction / N. A. Husaini, R. Ghazali, N. Mohd Nawi, L. H. Ismail // UCMA 2011, Part II. CCIS. — Springer, Heidelberg. — 2011. — Vol. 151. — P. 547–558.
109. Despotovic V. One-parameter fractional linear prediction / V. Despotovic, T. Skovranek, Z. Peric // Comput. Electr. Eng. Spec. Issue Signal Process. — 2018. — Vol. 69. — P.158–170.

110. Landassuri-Moreno V. Single-Step-Ahead and Multi-Step-Ahead Prediction with Evolutionary Artificial Neural Networks / V. Landassuri-Moreno, C. Bustillo-Hernández, J. Carbajal-Hernández, L. Fernández // Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2013. — Lecture Notes in Computer Science, Springer, Berlin, Heidelberg. — 2013. — Vol. 8258. — P.65-72. DOI: 10.1007/978-3-642-41822-8
111. Li Y. A new method of mapping relations from data based on artificial neural network / Y. Li, J. Liu, Q. Bao, W. Xu, R. Sadiq, and Y. Deng // International Journal of System Assurance Engineering and Management. — Springer. — 2014. — Vol. 5. — P. 544-553.
112. Stastny J. Analysis of Algorithms for Radial Basis Function Neural Network / J. Stastny and V. Skorpil // The International Federation for Information Processing (IFIP). — Springer, Boston, MA. — 2007. — Vol. 245. — P. 54-62.
113. Schwenker F. Three learning phases for radial-basis-function networks / F. Schwenker, H. A. Kestler, G. Palm // Neural Networks. — 2001. — No. 14. — P. 439–458. DOI:10.1016/s0893-6080(01)00027-2.
114. Kruse R. Computational Intelligence: A Methodological Introduction / R. Kruse, C. Borgelt, F. Klawonn, C. Moewes, and M. Steinbrecher // Springer, London. — 2016. — 563 p. DOI: 10.1007/978-1-4471-7296-3_1
115. Tkachenko R. Neurolike networks on the basis of Geometrical Transformation Machine / R. Tkachenko, I. Yurchak, U. Polishchuk // International Conference on Perspective Technologies and Methods in MEMS Design. –2008. – P. 77–80.
116. Tkachenko R. Model and Principles for the Implementation of Neural-Like Structures Based on Geometric Data Transformations / R. Tkachenko, I. Izonin // Advances in Computer Science for Engineering and Education. — Springer, Cham. — 2018. — Vol. 754. — P. 578–587.
117. Ткаченко Р.О. Побудова емпіричних формул за допомогою багатошарових нейроподібних структур геометричних перетворень / Р. О. Ткаченко, С. М. Дем'янчук // Науковий вісник НЛТУ України. — 2015. — Вип. 25.3. — С. 359–364.

118. R. Tkachenko, A. Doroshenko, I. Izonin, Y. Tsymbal, and B. Havrysh, “Imbalance Data Classification via Neural-like Structures of Geometric Transformations Model: Local and Global Approaches” In: Z.B. Hu, S. Petoukhov (eds) Advances in Computer Science for Engineering and Education (ICCSEEA2018), Advances in Intelligent Systems and Computing, Springer, Cham, 2018.
119. U. Polishchuk, P. Tkachenko, R. Tkachenko, I. Yurchak, Features of the auto-associative neurolike structures of the geometrical transformation machine (GTM), 5th International Conference on Perspective Technologies and Methods in MEMS Design, Zakarpattya, 2009, pp. 66–67.
120. R. Tkachenko, I. Yurchak, and U. Polishchuk, “Neurolike networks on the basis of Geometrical Transformation Machine”, 2008 International Conference on Perspective Technologies and Methods in MEMS Design, Polyana, 2008, pp. 77–80. DOI: 10.1109/MEMSTECH.2008.4558743
121. Деркач О. І. Аналітична обробка текстової інформації за допомогою засобів кластеризації // Науковий журнал «Молодий вчений»: Фізико-математичні науки. — 2016. — №7(34). — С. 159–165.
122. Хайкин С. «Нейронные сети: Полный курс», Neural Networks: A Comprehensive Foundation. — 2-е изд. — М.: «Вильямс», 2006. — 1104 с.
123. Hu, Zh., Bodyanskiy, Ye., Tyshchenko, O., Boiko, O.:A Neuro-Fuzzy Kohonen Network for Data Stream Possibilistic Clustering and Its Online Self Learning Procedure. Applied Soft Computing, Vol. 68, pp.710–718, (2018). DOI: 10.1016/j.asoc.2017.09.042
124. Волосюк Ю.В. Аналіз алгоритмів кластеризації для задач інтелектуального аналізу даних // Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. — 2014. — Вип. 47. — С. 112–119.
125. Колесницький О. К. Дослідження впливу типу метрики на точність кластеризації нейронною мережею кохонена у задачі медичного діагностування за аналізом крові / О. К. Колесницький, О. О. Журавська //

Інформаційні технології та комп’ютерна інженерія. — 2014. — № 3. — С. 6–11.

126. Bodyanskiy, Ye., Tyshchenko, O., Kopaliani, D.: An evolving connectionist system for data stream fuzzy clustering and its online learning. Neurocomputing. — 2017. — Vol. 262. — P. 41–56. DOI: 10.1016/j.neucom.2017.03.081
127. Переїденко А. В. Дослідження алгоритмів проведення кластерного аналізу для вирішення задач неруйнівного контролю / А. В. Переїденко, В. С. Єременко // Восточно-Европейский журнал передовых технологий. — 2010. — № 1/5(43). — С. 40–43.
128. Hocke, J., and Martinetz, T. (2013) "Feature weighting by maximum distance minimization" International Conference on Artificial Neural Networks, Springer Berlin Heidelberg.
129. Wu J. Advances in K-means Clustering // Springer Theses. — Berlin, Heidelberg. — 2012. — P. 1–16. DOI: 10.1007/978-3-642-29807-3.
130. Якимець Р.В. Методи кластеризації та їх класифікація // Міжнародний науковий журнал. — 2016. — No. 6 (2). — С. 48–50.
131. Hassanat A. B. A. Furthest-pair-based binary search tree for speeding big data classification using K-nearest neighbors // Big Data. — 2018. — No. 6(3). — P. 225–235. DOI: 10.1089/big.2018.0064.
132. Lange T. Stabilitybased validation of clustering solutions / T. Lange, V. Roth, M. L. Braun, J. M. Buhmann // Neural Computation. — 2004. — Vol. 16. — No. 6. — P. 1299–1323.
133. V. Pohrebennyk, O. Korchenko, O. Mitryasova, N. Bernatska, M. Kordos, “An analytical decision support system in prognostication of surface water pollution indicators”, 19th International Multidisciplinary Scientific Geoconference, SGEM 2019, Albena, Bulgaria, SGEM 2019, Vol. 19, Issue 2.1, pp. 49–56.
134. Мартовицький В. О. Класифікація методів виявлення аномалій в інформаційних системах / В. О. Мартовицький, І. В. Рубан, С. О. Партика. // Системи озброєння і військова техніка. — 2016. — №3. — С. 100–105.

135. D. Mekala, V. Gupta, B. Paranjape, and H. Karnick, “SCDV : Sparse composite document vectors using soft clustering over distributional representations,” in Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 659– 669.
136. M. Meila, “How to tell when a clustering is (approximately) correct using convex relaxations,” in Advances in Neural Information Processing Systems 31, 2018, pp. 7418–7429.
137. M. Kern, A. Lex, N. Gehlenborg, and C. R. Johnson, “Interactive visual exploration and refinement of cluster assignments,” BMC Bioinformatics, vol. 18, no. 1, Sep. 2017.
138. Kazarian, A., Teslyuk, V., Tsmots, I., and Mashevska, M. (2017) "Units and structure of automated “smart” house control system using machine learning algorithms" 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), pp. 364–366.
139. I. S. Dhillon, Y. Guan, and B. Kulis, “Kernel k-means: spectral clustering and normalized cuts,” in Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2004, pp. 551–556.
140. Martovytskyi V. Designing a monitoring model for cluster super-computer / V. Martovytskyi, I. Ruban, N. Lukova-Chuiko. // Eastern-European Journal of Enterprise Technologies. — 2016. — №84. — P. 32–37.
141. Розен В. П. Використання методу k-середніх кластертного аналізу під час розв’язання задач енергетичної безпеки територій / В. П. Розен, П. П. Іщук, Л. В. Давиденко // Науковий вісник Одеського національного економічного університету. — 2016. — №1. — С. 40–55.
142. Yoh-han Pao Adaptive Pattern Recognition and Neural Networks / Pao Yoh-han // Reading Massachusets. — Addison. — Wesley, 1989. — 309 p.
143. De Vito S. CO, NO2 and NOx urban pollution monitoring with on-field calibrated electronic nose by automatic Bayesian regularization / S. De Vito, M. Piga, L. Martinotto, G. Di Francia // Sensors and Actuators B: Chemical. — 2009. — Vol. 143, No. 1. — P. 182–191.

144. Takens F. Detecting strange attractors in turbulence / Dynamical Systems and Turbulence // Eds. D.Rang and L.S.Young. Dynamical Systems and Turbulence, Lecture Notes in Mathematics. — Springer-Verlag. — 1981. — Vol. 898. — P. 366–381.
145. Григоренко В.В., Еськов В.М. Анализ временных рядов в исследовании процессов хаотической динамики // Естественные и технические науки. — 2016. — №7. — С. 92–98.
146. Martovytskyi V. Investigation of network infrastructure control parameters for effective intellectual analysis / V. Martovytskyi, K. Smelyakov, D. Pribylnov, A. Chupryna // IEEE 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET), 20-24 Feb. 2018. – P. 983–986. DOI: 10.1109/TCSET.2018.8336359
147. I. Izonin, M. Greguš, R. Tkachenko, M. Logoida, O. Mishchuk, Y. Kynash, “SGD-based Wiener Polynomial Approximation for Missing Data Recovery in Air Pollution Monitoring Dataset”, In: Rojas I., Joya G., Catala A. (eds) Advances in Computational Intelligence. IWANN 2019. Lecture Notes in Computer Science, vol 11506, 2019, Springer, Cham, pp. 781–793.
148. Palamar M. Synthesis and optimization of neural network parameters for control of non-linear objects / M. Palamar, M. Aleksander, V. Pohrebennyk, M. Strembickyy // Przeglad Elektrotechniczny. — Volume 90, Issue 5. — 2014. — P. 207–210. DOI: 10.12915/pe.2014.05.47
149. “Спостереження за забрудненням атмосферного повітря в м. Києві” [Електронний ресурс] – Режим доступу: http://cgosreznevskyi.kiev.ua/index.php?fn=lsza&f=lsza
150. Ткаченко Р.О. Нейромережеві засоби штучного інтелекту: навчальний посібник / Р.О. Ткаченко, П.Р. Ткаченко, І.В. Ізонін, — Львів: Видавництво Львівської політехніки, 2017. — 240 с.
151. Su F. Y. Combining Hopfield neural network and contouring methods to enhance super–resolution mapping / Y. F. Su, G. M. Foody, A. M. Muad, and K.–S. Cheng

// IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. – 2012. – Vol. 5, № 5. – P. 1403–1417.

152. Бідюк П. І. Методи прогнозування / П. І. Бідюк, О. С. Меняйленко, О. В. Половцев. — Луганськ: Луганський національний ун-т ім. Тараса Шевченка, 2008. — Т. 1. — 305 с.
153. Ткаченко Р.О. Нейроподібні структури машини геометричних перетворень у завданнях інтелектуального аналізу даних / Р. Ткаченко, А. Дорошенко // // Вісник Національного університету “Львівська політехніка”. — 2009. — Комп’ютерні науки та інформаційні технології. — № 638. — С. 179–184.
154. Дорошенко А.В. Нейроподібні структури машини геометричних перетворень у завданнях інтелектуального аналізу даних /А. В. Дорошенко, Р. О. Ткаченко // Міжнародна наукова конференція «Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту» ISDMCI’2009: зб. наук. пр. у 2 т., 18–22 трав. 2009 р., Євпаторія, Україна. — Х.; Херсон, 2009. — Т. 2. — С. 309–314.
155. Izonin I. Learning-based image super-resolution using weight coefficients of synaptic connections / Ivan Izonin, Roman Tkachenko, Dmytro Peleshko, Taras Rak, Danylo Batyuk // Computer science and information technologies: proc. of X intern. scien. and techn. conf., 14-17 Sep. 2015 y., Lviv, Ukraine. – Lviv: Lviv Polytechnic Publishing House, 2015. — P. 25–29.
156. I. Izonin, R. Tkachenko, N. Kryvinska, P. Tkachenko, M. Greguš, “Multiple Linear Regression based on Coefficients Identification using Non-Iterative SGTM Neural-Like Structure”, In: Rojas I., Joya G., Catala A. (eds) Advances in Computational Intelligence, IWANN 2019, Lecture Notes in Computer Science, vol 11506, 2019, Springer, Cham, pp. 467–479.
157. Ізонін І.В. Метод збільшення роздільної здатності зображень на основі штучних нейронних мереж / І.В. Ізонін, Р.О. Ткаченко, Д.Д. Пелешко, Д.А. Батюк // Вісник Львівського державного університету безпеки життєдіяльності. — 2015. — № 11. — С. 47–56.

158. Guttag, John V. Introduction to Computation and Programming Using Python: With Application to Understanding Data. — MIT Press. — 2016.
159. Kenneth J Goldman An interactive environment for beginning Java programmers // Science of Computer Programming. — 2004. — Volume 53, Issue 1. — P. 3–24. DOI: 10.1016/j.scico.2004.02.002
160. Luca Ardito Effectiveness of Kotlin vs. Java in android app development tasks / Luca Ardito, Riccardo Coppola, Giovanni Malnati, Marco Torchiano // Information and Software Technology. – Volume 127. – 2020. DOI: 10.1016/j.infsof.2020.106374
161. E. Boj, T. Costa, J. Fortiana "Prediction Error in Distance-Based Generalized Linear Models", In: Palumbo F., Montanari A., Vichi M. (eds) Data Science. Studies in Classification, Data Analysis, and Knowledge Organization, Springer, Cham, 2017, pp. 191–204. DOI: 10.1007/978-3-319-55723-6_15

ДОДАТОК А

СПИСОК ПУБЛІКАЦІЙ ЗА ТЕМОЮ ДИСЕРТАЦІЙНОЇ РОБОТИ

1. Mishchuk O. Development of the method of forecasting the atmospheric air pollution parameters based on error correction by neural-like structures of the model of successive geometric transformations // Technology Audit and Production Reserves. — 2019. — № 6/2(50). — P. 22–26.
2. Mishchuk O. The Accelerated Method of Filling Gaps in Data Using a Linear SGTM Neural-Like Structure / O. Mishchuk, R. Tkachenko, V. Pohrebennyk // International Journal of Science and Engineering Investigations (IJSEI). – 2019. – №8(91). – P. 154-159.
3. Міщук О. С. Нейронна мережа з комбінованою апроксимацією поверхні відгуку / О. С. Міщук, П. Б. Вітинський // Наукові вісті КПІ: міжнародний науково-технічний журнал. – 2018. – № 2. – С. 18-24.
4. Міщук О. С. Методи оброблення та заповнення пропущених параметрів у даних екологічного моніторингу / О. С. Міщук, Р. О. Ткаченко // Науковий вісник НЛТУ України. – 2019. – №29(6). – С. 119-122. doi: 10.15421/40290623
5. Міщук О. С. Багатокрокове прогнозування тренду показників забруднення атмосферного повітря // Науковий вісник НЛТУ України. – 2019. – №29(8). – С. 142-146.
6. Mishchuk O. Missing Data Imputation Through SGTM Neural-Like Structure for Environmental Monitoring Tasks / O. Mishchuk, R. Tkachenko, I. Izonin // Advances in Computer Science for Engineering and Education II. – International Conference on Computer Science, Engineering and Education Applications, ICCSEEA 2019. – Springer. – Vol. 938. – P. 142-151. doi: 10.1007/978-3-030-16621-2_13 (***Scopus***)
7. Izonin I. SGD-Based Wiener Polynomial Approximation for Missing Data Recovery in Air Pollution Monitoring Dataset / I. Izonin, M. Greguš, R. Tkachenko, M. Logoyda, O. Mishchuk, Y. Kynash // Advances in Computational

Intelligence. – 15th International Work-Conference on Artificial Neural Networks, IWANN 2019. – Springer. – Vol. 11506. – P. 781-793. doi: 10.1007/978-3-030-20521-8_64 (***Scopus***)

8. Tkachenko R. A Non-Iterative Neural-Like Framework for Missing Data Imputation / R. Tkachenko, O. Mishchuk, I. Izonin, N. Kryvinska, R. Stoliarchuk // Procedia Computer Science. – The 14th International Conference on Future Networks and Communications, FNC 2019. – Springer. – Vol. 155. – P. 319-326. doi: 10.1016/j.procs.2019.08.046 (***Scopus***)
9. Izonin I. Recovery of Incomplete IoT Sensed Data using High-Performance Extended-Input Neural-Like Structure / I. Izonin, R. Tkachenko, N. Kryvinska, K. Zub, O. Mishchk, T. Lisovych // Procedia Computer Science. — International Workshop on Digitalization and Servitization within Factory-Free Economy, D&SwFFE-2019. — Elsevier. — Vol. 160. — P. 521-526. (***Scopus***)
10. Міщук О. С. Нейроподібні структури моделі геометричних перетворень з комбінованою апроксимацією поверхні відгуку // Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту (ISDMCI-2018) : матеріали XIV-ї міжнародної наукової конференції, Залізний Порт, 21-27 травня 2018. – Херсон. – С. 87-89.
11. Міщук О. С. Нелінійне розширення входів нейронної структури моделі послідовних геометричних перетворень // Обчислювальні методи і системи перетворення інформації (ОМІСПІ-2018) : збірник праць V-ї науково-технічної конференції, Львів, 4-5 жовтня 2018. – Львів. – С. 126-129.
12. Міщук О. С. Відновлення пропусків у даних моніторингу забруднення повітря за допомогою нейронної структури моделі послідовних геометричних перетворень // Комп'ютерне моделювання та оптимізація складних систем (КМОСС-2018) : матеріали IV-ї міжнародної науково-технічної конференції, Дніпро, 1-2 листопада 2018. – Дніпро. – С. 260-261.
13. Mishchuk O. Expansion of Neural-like structures inputs using combined approximation / O. Mishchuk, R. Tkachenko // Computer and Information

Systems and Technologies (CSITIC-2019) : Proceedings of the IIIrd International Scientific Conference, Kharkiv, 23-24 April 2019. – Kharkiv. – P. 29-32.

14. Mishchuk O. One-step Prediction of Air Pollution Control Parameters using Neural-Like Structure Based on Geometric Data Transformations / O. Mishchuk, R. Tkachenko // Electronics and Information Technologies (ELIT-2019) : Proceedings of the XIth International Scientific and Practical Conference, Lviv, 16-18 September 2019. – Lviv. – P. 192-197.
15. Міщук О. С. Прогнозування параметрів забруднення атмосферного повітря за допомогою лінійних нейроподібних структур // Побудова інформаційного суспільства: ресурси і технології: матеріали XVIII-ї міжнародної науково-практичної конференції, Київ, 19-20 вересня 2019. – Київ. – С. 275-278.
16. Mishchuk O. Neural network method of forecasting the air pollution trend by carbon monoxide // Information Technologies and Automation (ITA-2019) : Proceedings of the XIIth International Scientific Conference, Odesa, 17-18 October, 2019. – Odesa. – P. 101-102.
17. Міщук О. С. Підвищення точності прогнозування параметрів забруднення повітря // Комп'ютерне моделювання та оптимізація складних систем (КМОСС-2019) : матеріали V-ї міжнародної науково-технічної конференції, Дніпро, 6-8 листопада 2019. – Дніпро. – С.129-130.

ДОДАТОК Б

ТАБЛИЦІ, ВИКОРИСТАНІ У РОБОТІ

Таблиця 1.1.

Викиди ЗР в атмосферне повітря за 2013-2015 роки

Назва забруднюючої речовини	**2013 рік**	**2014 рік**	**2015 рік**
1. Викиди забруднюючих речовин, усього, тис. т	**277,3**	**252,1**	**203,6**
у тому числі від:			
1.1. стаціонарних джерел:	111,9	96,2	78,1
оксид вуглецю	3,6	3,4	4,5
діоксид та інші сполуки сірки	56,5	44,5	35,9
оксиди азоту	0,2	-	0,2
речовини у вигляді суспендованих твердих частинок	21,9	22,0	14,7
леткі органічні сполуки	1,2	1,1	1,0
1.2. пересувних джерел:	165,4	155,9	125,5
сірчистий ангідрид	2,0	1,8	1,5
оксиди азоту	0,1	-	-
оксид вуглецю	123,4	117,0	94,3
леткі органічні сполуки	18,4	17,5	13,6
речовини у вигляді суспендованих твердих частинок	2,5	2,2	1,8
1.2.1. автотранспорту:	156,3	147,6	117,4
діоксид та інші сполуки сірки	1,7	1,5	1,2
оксид азоту	0,1	-	-
оксид вуглецю	118,7	112,7	90,2
леткі органічні сполуки	17,4	16,6	12,8
речовини у вигляді суспендованих твердих частинок	2,2	1,9	1,5
2. Парникові гази, усього, млн. т. CO_2 – екв.	**8,7**	**7,7**	**6,2**

Таблиця 1.2.

Викиди найпоширеніших ПЗб в атмосферне повітря, тис. тонн

Назва параметру забруднення	**2000**	**2005**	**2010**	**2011**	**2012**	**2013**	**2014**	**2015**
Діоксид сірки	32,5	29,7	51,9	57,4	71,0	58,5	46,3	37,2
Діоксид азоту	17,4	19,8	32,5	35,4	40,8	36,4	31,3	24,1
Оксид азоту	…	0,3	0,3	0,3	0,3	0,3	0,3	0,2
Оксид вуглецю	79,8	82,7	125	126	137	127	120	98,8
Неметанові леткі органічні сполуки	1,1	3,4	20,8	20,7	21,9	19,5	18,6	14,6
Метан	12,3	15,4	15,1	12,5	10,2	10,5	10,4	11,0
Аміак	0,0	1,1	0,3	0,4	0,5	0,5	0,4	0,7

Таблиця 1.3.

ГДК параметрів забруднення атмосферного повітря

№ з/п	Найменування речовини	Величина максимально разової ГДКм.р., мг/м3	Величина середньо-добової ГДКс.д., мг/м3	Клас небезпеки речовини
1	Завислі речовини	0,5	0,15	3
2	Діоксид сірки	0,5	0,05	3
3	Оксид вуглецю	5,0	3,0	4
4	Діоксид азоту	0,20	0,04	3
5	Оксид азоту	0,40	0,06	3
6	Фтористий водень	0,02	0,005	2
7	Хлористий водень	0,2	0,2	2
8	Аміак	0,2	0,04	4
9	Формальдегід	0,035	0,003	2

ДОДАТОК В

ФРАГМЕНТИ КОДУ РОЗРОБЛЕНОГО ПРОГРАМОГО ЗАСОБУ

Підключення бібліотек:

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">

        <modelVersion>4.0.0</modelVersion>

    <groupId>com.mishchuk</groupId>
    <artifactId>forecasting</artifactId>
    <version>1.0-SNAPSHOT</version>


    <dependencies>
        <dependency>
            <groupId>net.sf.opencsv</groupId>
            <artifactId>opencsv</artifactId>
            <version>2.3</version>
        </dependency>
        <dependency>
            <groupId>org.apache.commons</groupId>
            <artifactId>commons-collections4</artifactId>
            <version>4.0</version>
        </dependency>
        <dependency>
            <groupId>commons-io</groupId>
            <artifactId>commons-io</artifactId>
            <version>2.6</version>
        </dependency>
        <dependency>
            <groupId>org.python</groupId>
            <artifactId>jython-slim</artifactId>
            <version>2.7.2b3</version>
        </dependency>

    </dependencies>

    <build>
        <plugins>
            <plugin>
                <artifactId>maven-compiler-plugin</artifactId>
                <configuration>
                    <source>1.8</source>
                    <target>1.8</target>
                </configuration>
            </plugin>
        </plugins>
    </build>

</project>
```

Взаємодія з Pyhton-фреймворком:

```
package com.mishchuk.python;

import java.io.InputStream;
import java.util.ArrayList;
import java.util.List;

import org.python.antlr.ast.Str;
import org.python.core.Py;
import org.python.core.PyArray;
import org.python.core.PyFloat;
import org.python.core.PyObject;
import org.python.core.PyString;
import org.python.util.PythonInterpreter;


public class PythonWrapper
{
   private static final String PYTHON_SCRIPT_NAME = "neural_networks.py";
   private static PythonInterpreter interpreter = new PythonInterpreter();

   public enum PythonAlgorithm
   {
      ADAPTIVE_BOOSTING("adaptive_boosting"),
      SGT_MODEL("sgt_model"),
      DECISION_TREE("decision_tree"),
      LINEAR_GRADIENT_DESCENT("linear_gradient_descent"),
      MULTILAYER_PERCEPTRON("multilayer_perceptron"),
      NATIVE_PREDICT("native_predict"),
      NEAREST_NEIGHBORS("nearest_neighbors"),
      RANDOM_FOREST("random_forest"),
      SUPPORT_VECTOR_REGRESSION("support_vector_regression");

      private String pythonModelName;
      PythonAlgorithm(final String pythonModelName)
      {
         this.pythonModelName = pythonModelName;
      }

      public String getPythonModelName()
      {
         return pythonModelName;
      }
   }

   public static List<List<Double>> executeAlgorithm(final PythonAlgorithm
algorithm, final List<List<Double>> input)
   {
      final InputStream pyCodeStream =
PythonWrapper.class.getClassLoader().getResourceAsStream(PYTHON_SCRIPT_NAME);

      final PyArray inputArray = toPyArray(input);

      interpreter.set("input", input);
      interpreter.set("model", new PyString(algorithm.getPythonModelName()));
      interpreter.execfile(pyCodeStream);
      final PyArray outputArray = (PyArray) interpreter.get("output");

      return fromPyArray(outputArray);
   }

   private static PyArray toPyArray(final List<List<Double>> input)
   {
```

```
        final PyArray inputArray = new PyArray(PyArray.class, input.size());
        for(final List<Double> row : input)
        {
            final PyArray inputRow = new PyArray(PyFloat.class, row.size());

            for(final Double val : row)
            {
                final PyFloat pyVal = new PyFloat(val);
                inputArray.__add__(pyVal);
            }
            inputArray.__add__(inputRow);
        }

        return inputArray;
    }


    private static List<List<Double>> fromPyArray(PyArray outputArray)
    {
        final List<List<Double>> result = new ArrayList<>();
        int rowCount = outputArray.getItemsize();
        for(int i = 0; i < rowCount; i++)
        {
            final List<Double> row = new ArrayList<>();
            final PyArray pyRow = outputArray.__get__(i, PyArray.class)

            int colCount =

            final PyArray inputRow = new PyArray(PyFloat.class, row.size());

            for(final Double val : row)
            {
                final PyFloat pyVal = new PyFloat(val);
                inputArray.__add__(pyVal);
            }
            inputArray.__add__(inputRow);
        }
    }
}
```

Передобробка даних:

```
package com.mishchuk.preprocessing;

import java.util.ArrayList;
import java.util.List;


public class Normalization
{
    public static List<Double> getColumnMaximums(final List<List<Double>> input)
    {
        final List<Double> maxValues = new ArrayList<>(input.get(0).size());
        maxValues.addAll(input.get(0));

        for (final List<Double> row : input)
        {
            for (int i = 0; i < row.size(); i++)
            {
                final Double val = Math.abs(row.get(i));
                if (val > maxValues.get(i))
```

```
            {
               maxValues.set(i, val);
            }
         }
      }

      return maxValues;
   }

   public static List<List<Double>> normalizeColumns(final List<List<Double>>
input)
   {
      final List<Double> maxValues = getColumnMaximums(input);
      return Scaling.scaleColumns(input, maxValues);
   }

   public static List<List<Double>> normalizeRows(final List<List<Double>> input)
   {
      final List<Double> maxValues = new ArrayList<>(input.size());
      for (final List<Double> row : input)
      {
         maxValues.add(row.get(0));
      }

      for (int i = 0; i < input.size(); i++)
      {
         final List<Double> row = input.get(i);
         for (final Double v : row)
         {
            final Double val = Math.abs(v);
            if (val > maxValues.get(i))
            {
               maxValues.set(i, val);
            }
         }
      }

      return Scaling.scaleRows(input, maxValues);
   }

   public static List<List<Double>> normalizeTable(final List<List<Double>>
input)
   {
      double maxValue = input.get(0).get(0);

      for (final List<Double> row : input)
      {
         for (final Double v : row)
         {
            final Double val = Math.abs(v);
            if (val > maxValue)
            {
               maxValue = val;
            }
         }
      }

      return Scaling.scaleTable(input, maxValue);
   }
}
```

```java
package com.mishchuk.preprocessing;

import java.util.ArrayList;
import java.util.List;


public class Scaling
{
   public static List<List<Double>> scaleColumns(final List<List<Double>> input,
final List<Double> scaleFactors)
   {
      final List<List<Double>> result = new ArrayList<>();
      for (final List<Double> row : input)
      {
         final List<Double> scaledRow = new ArrayList<>();
         for (int i = 0; i < row.size(); i++)
         {
            scaledRow.add(row.get(i) / scaleFactors.get(i));
         }
         result.add(scaledRow);
      }

      return result;
   }

   public static List<List<Double>> scaleRows(final List<List<Double>> input,
final List<Double> scaleFactors)
   {
      final List<List<Double>> result = new ArrayList<>();
      for (int i = 0; i < input.size(); i++)
      {
         final List<Double> row = input.get(i);
         final List<Double> scaledRow = new ArrayList<>();
         for (int j = 0; j < row.size(); j++)
         {
            scaledRow.add(row.get(j) / scaleFactors.get(i));
         }
         result.add(scaledRow);
      }

      return result;
   }

   public static List<List<Double>> scaleTable(List<List<Double>> input, double
scaleFactor)
   {
      final List<List<Double>> result = new ArrayList<>();
      for (final List<Double> row : input)
      {
         final List<Double> scaledRow = new ArrayList<>();
         for (final Double v : row)
         {
            scaledRow.add(v / scaleFactor);
         }
         result.add(scaledRow);
      }

      return result;
   }
}
```

Кластеризація:

```
package com.mishchuk.inputextension;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import com.mishchuk.preprocessing.Distance;
import com.mishchuk.preprocessing.Splitting;


public class Clusterization
{
   public static List<List<Double>> clusterizeTest(final List<List<Double>>
train, final List<List<Double>> testFull, final int clusterCount)
   {
      final Map<Integer, Integer> testToCluster = new HashMap<>();
      final int testColumnCount = testFull.get(0).size();
      final List<Double> testY = Splitting.extractColumn(testFull,
testColumnCount - 1);
      final List<List<Double>> test = new ArrayList<>();
      Splitting.copyTablePartByCol(testFull, test, 0, testColumnCount - 1);

      for (int z = 0; z < test.size(); z++)
      {
         final List<Double> testRow = test.get(z);
         final List<Double> distances = new ArrayList<>();

         for (List<Double> trainRow : train)
         {
            distances.add(Distance.getDistance(testRow, trainRow));
         }

         final List<Minimum> minimums = new ArrayList<>();

         for (int i = 0; i < clusterCount; i++)
         {
            Double min = distances.get(0);
            int index = 0;
            for (int j = 0; j < distances.size(); j++)
            {
               if (distances.get(j) < min)
               {
                  min = distances.get(j);
                  index = j;
               }
            }

            final Minimum m = new Minimum();
            m.setId(index);
            m.setDistance(min);
            m.setVector(train.get(index));
            minimums.add(m);

            distances.set(index, Double.MAX_VALUE);
         }

         final int cluster = getCluster(minimums);
         testToCluster.put(z + 1, cluster);
      }

      return generateClusters(test, testY, testToCluster);}
```

```
   public static List<List<Double>> clusterizeTrain(final List<List<Double>>
train, final List<List<Double>> testFull, final int clusterCount)
   {
      final Map<Integer, Integer> testToCluster = new HashMap<>();
      final int testColumnCount = testFull.get(0).size();
      final List<Double> testY = Splitting.extractColumn(testFull,
testColumnCount - 1);
      final int trainColumnCount = train.get(0).size();
      final List<Double> trainY = Splitting.extractColumn(train, trainColumnCount
- 1);
      final List<List<Double>> test = new ArrayList<>();
      Splitting.copyTablePartByCol(testFull, test, 0, testColumnCount - 1);

      for (int z = 0; z < test.size(); z++)
      {
         final List<Double> testRow = test.get(z);
         final List<Double> distances = new ArrayList<>();

         for (List<Double> trainRow : train)
         {
            distances.add(Distance.getDistance(testRow, trainRow));
         }

         final List<Minimum> minimums = new ArrayList<>();

         for (int i = 0; i < clusterCount; i++)
         {
            Double min = distances.get(0);
            int index = 0;
            for (int j = 0; j < distances.size(); j++)
            {
               if (distances.get(j) < min)
               {
                  min = distances.get(j);
                  index = j;
               }
            }

            final Minimum m = new Minimum();
            m.setId(index);
            m.setDistance(min);
            m.setVector(train.get(index));
            minimums.add(m);

            distances.set(index, Double.MAX_VALUE);
         }

         final int cluster = getCluster(minimums);
         testToCluster.put(z + 1, cluster);
      }

      return generateClusters(train, trainY, testToCluster);
   }

   private static int getCluster(List<Minimum> minimums)
   {
      final Map<Integer, Integer> counts = new HashMap<>();
      for (final Minimum minimum : minimums) {
         final int key = minimum.getId() + 1;
         Integer c = counts.get(key);
         if(c == null) {
            c = 0;
         }
```

```
            c++;
            counts.put(key, c);
         }

         return counts.entrySet().stream().sorted((e1, e2) ->
e2.getValue().compareTo(e1.getValue())).map(Map.Entry::getKey).findFirst().get();
      }

      private static List<List<Double>> generateClusters(List<List<Double>> test,
List<Double> testY, Map<Integer, Integer> testToCluster)
      {
         final List<List<Double>> result = new ArrayList<>();
         final int clusterCount =
testToCluster.values().stream().max(Integer::compare).get();
         for (int z = 0; z < test.size(); z++)
         {
            final List<Double> row = new ArrayList<>(test.get(0).size() +
clusterCount + 1);
            row.addAll(test.get(z));
            for(int i = 0; i < clusterCount; i++) {
               row.add(testToCluster.get(z + 1) == (i + 1) ? 1.0d : 0.0d);
            }
            row.add(testY.get(z));
            result.add(row);
         }
         return result;
      }

      private static class Minimum
      {
         private List<Double> vector;
         private int id;
         private Double distance;

         public List<Double> getVector()
         {
            return vector;
         }

         public void setVector(List<Double> vector)
         {
            this.vector = vector;
         }

         public int getId()
         {
            return id;
         }

         public void setId(int id)
         {
            this.id = id;
         }

         public Double getDistance()
         {
            return distance;
         }

         public void setDistance(Double distance)
         {
            this.distance = distance;
         } } }
```

Метод часових вікон:

```
package com.mishchuk.timeseries;

import java.util.ArrayList;
import java.util.List;
import java.util.concurrent.ThreadLocalRandom;
import java.util.stream.Collectors;

import com.mishchuk.preprocessing.Splitting;


public class TimeSeries
{
   public static List<List<List<Double>>> calculate(final List<List<Double>>
input, final int k)
   {
      final List<List<List<Double>>> output = new ArrayList<>();
      for (int i = 0; i < input.get(0).size(); i++)
      {
         final int finalI = i;
         final List<Double> col = input.stream().map(row ->
row.get(finalI)).collect(Collectors.toList());

         final List<List<Double>> newTable = new ArrayList<>();

         for (int j = 0; j < (col.size() - k - 1); j++)
         {
            final List<Double> newRow = new ArrayList<>(k + 1);
            Splitting.copyCollection(col, newRow, j, k + 1);
            newTable.add(newRow);
         }

         final List<Double> finalRow = new ArrayList<>(k + 1);
         Splitting.copyCollection(col, finalRow, col.size() - k - 1, k);
         finalRow.add(col.get(ThreadLocalRandom.current().nextInt(col.size())));
         newTable.add(finalRow);

         for (List<Double> row : newTable)
         {
            row.add(row.get(row.size() - 1));
         }

         output.add(newTable);
      }
      return output;
   }
}
```

Пришвидшення прогнозування парамтерів зібруднення атмосферного повітря за рахунок поєднання НС МПГП та лінійного полінома:

```
package com.mishchuk.acceleration;

import java.util.ArrayList;
import java.util.List;

import com.mishchuk.preprocessing.Splitting;


public class LinearPolynomial
{
   public static List<List<Double>> calculate(final List<List<Double>> input,
final List<List<Double>> prediction,
         final List<List<Double>> error)
   {
      final List<Double> inputCol = Splitting.extractColumn(input, 2);
      final List<Double> coefficients = new ArrayList<>();

      coefficients.add(inputCol.get(0));
      for (int i = 1; i < inputCol.size(); i++)
      {
         coefficients.add(inputCol.get(i) - inputCol.get(0));
      }

      final List<Double> predictedOutputs = new ArrayList<>();

      final List<List<Double>> output = new ArrayList<>();

      for (int i = 0; i < prediction.size(); i++)
      {
         Double predictedOutput = coefficients.get(0);
         final List<Double> row = prediction.get(i);
         for (int j = 0; j < row.size(); j++)
         {
            predictedOutput += coefficients.get(j + 1) * row.get(j);
         }
         predictedOutputs.add(predictedOutput);
      }

      final List<Double> avg = new ArrayList<>();
      for (int i = 0; i < 5; i++)
      {
         avg.add(0.0d);
      }

      for (int i = 0; i < predictedOutputs.size(); i++)
      {
         final Double a = error.get(i).get(0);
         final Double b = predictedOutputs.get(i);

         final List<Double> row = new ArrayList<>();
         row.add(a);
         row.add(b);

         final Double diff = b - a;
         row.add(Math.abs(diff));
         avg.set(2, avg.get(2) + Math.abs(diff));

         row.add(diff * diff);
```

```
            avg.set(3, avg.get(3) + diff * diff);

            row.add(Math.abs(diff / a));
            avg.set(4, avg.get(4) + Math.abs(diff / a));

            output.add(row);
         }

         for (int i = 0; i < avg.size(); i++)
         {
            avg.set(i, avg.get(i) / predictedOutputs.size());
         }

         avg.set(3, Math.sqrt(avg.get(3)));
         avg.set(4, avg.get(4) * 100);

         output.add(avg);

         return output;
      }
}
```

Метод розширення входів за допомогою попереднього виділення КМТ:

```
package com.mishchuk.inputextension;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import com.mishchuk.preprocessing.Distance;
import com.mishchuk.preprocessing.Splitting;


public class KNN
{
   public static List<List<Double>> clusterize(final List<List<Double>> train,
final List<List<Double>> testFull, final int clusterCount)
   {
      final Map<Integer, Integer> testToCluster = new HashMap<>();
      final int testColumnCount = testFull.get(0).size();
      final List<Double> testY = Splitting.extractColumn(testFull,
testColumnCount - 1);
      final List<List<Double>> test = new ArrayList<>();
      Splitting.copyTablePartByCol(testFull, test, 0, testColumnCount - 1);

      for (int z = 0; z < test.size(); z++)
      {
         final List<Double> testRow = test.get(z);
         final List<Double> distances = new ArrayList<>();

         for (List<Double> trainRow : train)
         {
            distances.add(Distance.getDistance(testRow, trainRow));
         }

         //       List<Double> distancesOriginal = new ArrayList<>(distances);
         List<Minimum> minimums = new ArrayList<>();

         for (int i = 0; i < clusterCount; i++)
         {
            Double min = distances.get(0);
```

```
            int index = 0;
            for (int j = 0; j < distances.size(); j++)
            {
               if (distances.get(j) < min)
               {
                  min = distances.get(j);
                  index = j;
               }
            }

            Minimum m = new Minimum();
            m.setId(index);
            m.setDistance(min);
            m.setVector(train.get(index));
            minimums.add(m);

            distances.set(index, Double.MAX_VALUE);
         }

         final int cluster = getCluster(minimums);
         testToCluster.put(z + 1, cluster);
      }

      return generateTestClusters(test, testY, testToCluster);
   }

   private static int getCluster(List<Minimum> minimums)
   {
      final Map<Integer, Integer> counts = new HashMap<>();
      for (final Minimum minimum : minimums) {
         final int key = minimum.getId() + 1;
         Integer c = counts.get(key);
         if(c == null) {
            c = 0;
         }
         c++;
         counts.put(key, c);
      }

      return counts.entrySet().stream().sorted((e1, e2) ->
e2.getValue().compareTo(e1.getValue())).map(Map.Entry::getKey).findFirst().get();
   }

   private static List<List<Double>> generateTestClusters(List<List<Double>>
test, List<Double> testY, Map<Integer, Integer> testToCluster)
   {
      final List<List<Double>> result = new ArrayList<>();
      final int clusterCount =
testToCluster.values().stream().max(Integer::compare).get();
      for (int z = 0; z < test.size(); z++)
      {
         final List<Double> row = new ArrayList<>(test.get(0).size() +
clusterCount + 1);
         row.addAll(test.get(z));
         for(int i = 0; i < clusterCount; i++) {
            row.add(testToCluster.get(z + 1) == (i + 1) ? 1.0d : 0.0d);
         }
         row.add(testY.get(z));
         result.add(row);
      }
      return result;
   }

   private static class Minimum
```

```
    {
        private List<Double> vector;
        private int id;
        private Double distance;

        public List<Double> getVector()
        {
            return vector;
        }

        public void setVector(List<Double> vector)
        {
            this.vector = vector;
        }

        public int getId()
        {
            return id;
        }

        public void setId(int id)
        {
            this.id = id;
        }

        public Double getDistance()
        {
            return distance;
        }

        public void setDistance(Double distance)
        {
            this.distance = distance;
        }
    }
}
```

Метод корекції похибки:

```
package com.mishchuk.errorcorrection;

import java.util.ArrayList;
import java.util.List;

import com.mishchuk.errorcalculation.MAE;
import com.mishchuk.preprocessing.EuclidDistance;
import com.mishchuk.preprocessing.Gauss;
import com.mishchuk.preprocessing.Normalization;
import com.mishchuk.preprocessing.Splitting;


public class GRNN
{
    public static List<Double> calculateResponses(final List<List<Double>> train,
final List<List<Double>> test)
    {
        final List<List<Double>> trainScaled = Normalization.normalizeRows(train);
        final List<List<Double>> testScaled = Normalization.normalizeRows(test);

        final List<List<Double>> trainEuclidDistancesTable = new ArrayList<>();
        for(int i = 0; i < train.size(); i++)
        {
```

```
trainEuclidDistancesTable.add(EuclidDistance.getEuclidDistances(trainScaled.get(i
), trainScaled, i));
      }

      final List<Double> trainYs = Splitting.extractColumn(trainScaled,
trainScaled.get(0).size() - 1);
      List<Double> trainResponses = new ArrayList<>();
      double bestSigma = 0;
      double minimumError = Double.MAX_VALUE;
      for(double sigma = 0.05d; sigma <= 10d; sigma += 0.05d)
      {
         final List<List<Double>> trainGaussFunctionsTable =
Gauss.calculateGaussFunctionsTable(trainEuclidDistancesTable, sigma);

         final List<Double> trainResponsesNext = new ArrayList<>();
         for (int j = 0; j < trainScaled.size(); j++)
         {
            final List<Double> gaussFunctions = trainGaussFunctionsTable.get(j);
            int gaussIndex = 0;
            double sumUp = 0.0d;
            double sumDown = 0.0d;
            for (int i = 0; i < trainScaled.size(); i++)
            {
               if (i != j)
               {
                  sumUp += gaussFunctions.get(gaussIndex) *
getY(trainScaled.get(i));
                  sumDown += gaussFunctions.get(gaussIndex);
                  gaussIndex++;
               }
            }
            final double sumDownMin = Math.pow(10, -4);
            if (sumDown < sumDownMin)
            {
               sumDown = sumDownMin;
            }

            trainResponsesNext.add(sumUp / sumDown);
         }

         double error = MAE.calculate(trainResponsesNext, trainYs);
         if(error < minimumError)
         {
            trainResponses.clear();
            trainResponses.addAll(trainResponsesNext);
            bestSigma = sigma;
            minimumError = error;
         }
      }

      final List<List<Double>> testEuclidDistancesTable = new ArrayList<>();
      for(int i = 0; i < test.size(); i++)
      {

testEuclidDistancesTable.add(EuclidDistance.getEuclidDistances(testScaled.get(i),
trainScaled, -1));
      }

      final List<List<Double>> testGaussFunctionsTable =
Gauss.calculateGaussFunctionsTable(testEuclidDistancesTable, bestSigma);

      final List<Double> testResponses = new ArrayList<>();
      for(int j = 0; j < testScaled.size(); j++)
      {
```

```
            final List<Double> gaussFunctions = testGaussFunctionsTable.get(j);
            int gaussIndex = 0;
            double sumUp = 0.0d;
            double sumDown = 0.0d;
            for (int i = 0; i < testScaled.size(); i++)
            {
               sumUp += gaussFunctions.get(gaussIndex) * getY(trainScaled.get(i));
               sumDown += gaussFunctions.get(gaussIndex);
               gaussIndex++;
            }
            final double sumDownMin = Math.pow(10, -4);
            if(sumDown < sumDownMin)
            {
               sumDown = sumDownMin;
            }

            testResponses.add(sumUp / sumDown);
         }

         return testResponses;
      }

      private static double getY(final List<Double> row)
      {
         return row.get(row.size() - 1);
      }
}
```

ДОДАТОК Г

ОПИС НАЛАШТУВАННЯ ПАРАМЕТРІВ МОДЕЛЕЙ ПРОГНОЗУВАННЯ В УМОВАХ ПРОПУЩЕНИХ ДАНИХ

Модель ***SVR*** містить в собі такі параметри налаштування:

- *kernel* - dказує тип ядра, який буде використовуватися в алгоритмі, що повинен бути одним із ‘linear’, ‘poly’, ‘rbf’, ‘sigmoid’, ‘precomputed’ або для виклику (якщо нічого не вказано, буде використано "rbf", якщо вказано виклик, він використовується для попереднього обчислення матриці ядра);
- *degree* - ступінь функції полінома ядра ('poly') (усі інші ядра ігноруються);
- *gamma* - {‘scale’, ‘auto’} - коефіцієнт ядра для 'rbf', 'poly' та 'sigmoid' (якщо передано gamma = 'scale' (за замовчуванням), то воно використовує 1 / (n_features * X.var ()) як значення гамми; якщо "auto", 1 / n_features);
- *coef0* - незалежний термін у функції ядра (має значення лише в ‘poly’ та ‘sigmoid’);
- *tol* - допуск для зупинки критерію;
- *c* - параметр регуляризації. Сила регуляризації обернено пропорційна C;
- *epsilon* - визначає епсилон-трубку, в межах якої не виконується штраф у функції втрати тренувань з балами, передбаченими на відстані епсілона від фактичного значення;
- *shrinking* – параметр, що вказує чи використовувати евристику скорочення;
- *cache_size* – параметр, що визначає розмір кешу ядра (у МБ);
- *verbose* - параметр, що визначає чи включити докладний висновок (це налаштування використовує перевагу налаштування часу виконання в libsvm, яке, якщо воно включено, може не працювати належним чином у багатопотоковому контексті;
- *max_iter* - жорсткий ліміт ітерацій в межах розв'язувача або -1 без обмежень.

Налаштування моделі ***RandomForestRegressor*** містить в собі наступні параметри:

- *n_estimators* – число дерев лісу, котре по замовчуванню складає 10;
- *criterion* – функція для вимірювання якості розбиття, котра залежить від дерева та вибирається поміж {“mse”, friedman_mse”, “mae”};
- *max_depth* – число максимальної глибини дерева;
- *min_samples_split* – число мінімальної кількості зразків, необхідне для розділення внутрішнього вузла (по замовчуванню становить 1);
- *min_samples_leaf* – число мінімальної кількості зразків у новостворених листках (розкол зупиняється, якщо після розщеплення один із листків містить менше зразків ніж min_samples_leaf);
- *min_density* – параметр, що контролює контролює мінімальну фракції зразків у масці (якщо щільність опускається нижче цього порогового значення, маска перераховується, а вхідні дані упаковуються, що призводить до копіювання даних; якщо min_density дорівнює одиниці, розділи завжди представляються як копії вихідних даних);
- *max_features* – кількість функцій, які слід враховувати, шукаючи найкращий розділ;
- *bootstrap* – додатковий параметр, що вказує чи використовуються зразки завантажувальної машини при побудові дерев;
- *oob_score* – параметр, котрий вказує чи використовувати зразки для оцінки похибки узагальнення;
- *n_jobs* – кількість завдань, які потрібно виконати паралельно (якщо -1, то кількість завдань встановлюється рівною кількості ядер);
- *verbose* – контролює деталі процесу побудови дерев та по замовчуванню становить 0.

Модель ***DecisionTreeRegressor*** налаштовується такими парметрами:

- *criterion* – функція для вимірювання якості розбиття, котра залежить від дерева та вибирається поміж {“mse”, friedman_mse”, “mae”};
- *max_depth* – число максимальної глибини дерева;
- *min_samples_split* – число мінімальної кількості зразків, необхідне для розділення внутрішнього вузла (по замовчуванню становить 1);
- *min_samples_leaf* – число мінімальної кількості зразків у новостворених листках (розкол зупиняється, якщо після розщеплення один із листків містить менше зразків ніж min_samples_leaf);
- *max_features* – кількість функцій, які слід враховувати, шукаючи найкращий розділ;
- *splitter* – використовується для вибору стратегії розбиття на кожному вузлі. (“best" для вибору найкращого поділу та "random" для вибору найкращого випадкового розбиття);
- *min_weight_fraction_leaf* - Мінімальна зважена частка від загальної ваги (усіх вхідних зразків), необхідних для розташування у вузлі листка;
- *max_leaf_nodes* – параметр, що задає найкращі вузли як відносне зменшення домішок (якщо не вказаний, то буде необмежена кількість вузлів листя);
- *min_impurity_decrease* – параметр, що вказує чи вузол буде розщеплений, якщо цей розкол викликає зменшення домішки, що перевищує або дорівнює цьому значенню;
- *min_impurity_split* – параметр, що встановлю поріг для ранньої зупинки росту дерева (вузол розділяється, якщо його домішка перевищує поріг, інакше це лист);
- *ccp_alpha* – параметр, що використовується для зупинки росту дерева (Вибирається піддерево з найбільшою складністю витрат, меншою, ніж ccp_alpha).

Модель ***SGDRegressor*** має такі параметри налаштування процесу навчання та застосування:

- *loss* - {'squared_loss', 'huber', 'epsilon_insensitive', 'squared_epsilon_insensitive'} - параметр, що визначає функцію втрат;
- *penalty* – параметр, що є регулятором регуляризації вибору функції;
- *alpha* - константа, що підсилює регуляризацію та використовується для обчислення learning_rate;
- *l1_ratio* - параметр змішування Elastic Net з 0 <= l1_ratio <= 1;
- *fit_intercept* – вказує чи слід перехоплювати оцінку чи ні (якщо False, дані вважаються вже центрованими);
- *max_iter* - максимальна кількість пропусків за навчальними даними (впливає лише на поведінку методу fit, а не на метод part_fit);
- *tol* - критерій зупинки (якщо не None, ітерації припиняться, коли (втрата> best_loss - tol) протягом n_iter_no_change послідовних епох);
- *shuffle* – параметр, що визначає чи слід змішувати дані тренувань після кожної епохи;
- *epsilon* - епсилон у функціях втрати, нечутливих до епсилону; лише якщо втрата - "huber", "epsilon_insensitive" або "squared_epsilon_insensitive" (для "huber" визначає поріг, при якому стає менш важливим правильний прогноз);
- *learning_rates* - графік курсу навчання: 'constant': eta = eta0, 'optimal': eta = 1.0 / (alpha * (t + t0)), 'invscaling': eta = eta0 / pow(t, power_t), 'adaptive': eta = eta0, до тих пір, поки навчання не знижується (щоразу, коли n_iter_no_change послідовних епох не вдається зменшити втрату тренінгу на tol або не збільшити бал валідації на tol, якщо true_stopping є True, поточний рівень навчання ділиться на 5);

- *eta0* - початковий коефіцієнт навчання для "постійного", "запрошуючого" або "адаптивного" розкладу (значення за замовчуванням - 0,01);
- *power_t* - коефіцієнт зворотного масштабування швидкості навчання;
- *early_stopping* – вказує чи використовувати ранню зупинку для припинення навчання, коли оцінка перевірки не покращується (якщо встановлено значення True, воно автоматично відмінить частину даних тренувань як перевірку та припинить навчання, коли оцінка перевірки не покращиться принаймні для n_iter_no_change послідовних епох);
- *validation_fraction* - частка даних тренувань, які слід відкласти як набір перевірки для раннього припинення (повинно бути від 0 до 1 та використовується лише якщо true_stopping – True);
- *n_iter_no_change* - кількість повторень, які не мають покращення, дочекатися перед ранньою зупинкою;
- *warm_start* – неодноразовий виклик придатного або partly_fit, коли hot_star починається істинним, може призвести до іншого рішення, ніж при одночасному виклику фітнеру через спосіб переміщення даних (якщо встановлено значення True, використовується рішення попереднього дзвінка повторно, щоб він підходив як ініціалізація, інакше видаляється попереднє рішення; zкщо використовується динамічна швидкість навчання, швидкість навчання адаптується залежно від кількості вже проглянутих зразків, а виклик пристосування скидає цей лічильник, тоді як partly_fit призводить до збільшення наявного лічильника);
- *average* - якщо встановлено значення True, обчислює усереднені ваги SGD і зберігає результат у атрибуті coef_. (якщо встановлено інт, більший за 1, усереднення розпочнеться, коли загальна кількість розглянутих зразків досягне середнього).

Модель ***MLPRegressor*** налаштовується наступними параметрами:

- *hidden_layer_size* - і-й елемент представляє кількість нейронів у і-му прихованому шарі;
- *activation* - {‘identity’, ‘logistic’, ‘tanh’, ‘relu’} - функція активації для прихованого шару;
- *solver* - {‘lbfgs’, ‘sgd’, ‘adam’} - вирішувач для оптимізації ваги ("adam" за замовчуванням працює досить добре на відносно великих наборах даних з тисячами навчальних зразків і більше з точки зору як часу навчання, так і оцінки валідації, однак для невеликих наборів даних "lbfgs" конвергується швидше та краще;
- *batch_size* - розмір мініатюр для стохастичних оптимізаторів (якщо розв'язувачем є "lbfgs", класифікатор не використовує міні-пакет);
- *learning_rate* - {‘constant’, ‘invscaling’, ‘adaptive’} - графік курсу навчання для оновлення ваги, де " *constant*’" - це постійна швидкість навчання, задана "learning_rate_init", " *invscaling*’" поступово знижує швидкість навчання learning_rate_ на кожному кроці "t", використовуючи зворотний показник масштабування "power_t", а " *adaptive*’" підтримує постійну швидкість навчання до "learning_rate_init" до тих пір, поки втрати в навчанні не зменшуються (кожен раз, коли дві послідовні епохи не вдається зменшити втрату тренінгу принаймні на tol або не збільшити оцінку валідації принаймні на tol, якщо увімкнено функцію «раннє припинення», поточний рівень навчання ділиться на 5);
- *learning_rate_init* - використовувана початкова норма навчання (кконтролює розмір кроків при оновленні ваг та використовується лише тоді, коли solver = 'sgd' або 'adam');
- *power_t* - показниу ступеня зворотного масштабування, що використовується при оновленні ефективного коефіцієнта навчання, коли

налаштування_набору встановлено на "запрошення" (використовується лише тоді, коли solver = 'sgd');

- *max_iter* - максимальна кількість ітерацій;
- *shuffle* – параметр, що вказує чи слід перемішувати зразки в кожній ітерації (використовується лише тоді, коли solver = 'sgd' або 'adam');
- *tol* - толерантність до оптимізації (якщо втрата чи оцінка не покращуються, принаймні, tol для послідовних ітерацій n_iter_no_change, якщо тільки у навчальному_rate не встановлено "адаптивний", конвергенція вважається досягнутою, а навчання припиняється);
- *verbose* – параметр, що вказує чи надрукувати повідомлення про прогрес у stdout;
- *warm_start* - якщо встановлено значення True, використовується рішення попереднього дзвінка повторно, щоб він підходив як ініціалізація.;
- momentum - момент для оновлення градієнта спуску (має бути від 0 до 1 та Використовується лише тоді, коли solver = 'sgd');
- *nesterovs_momentum* – параметр, що вказує чи використовувати імпульс Нестерова (використовується лише коли solver = 'sgd' та імпульс> 0);
- *early_stopping* - парамер, що вказує чи використовувати ранню зупинку для припинення навчання, коли оцінка перевірки не покращується (якщо встановлено значення true, воно автоматично відкладе 10% даних тренувань як валідацію та припинить навчання, коли оцінка перевірки не покращиться принаймні для n_iter_no_change послідовних епох. Діє лише тоді, коли solver = 'sgd' або 'adam');
- *validation_fraction* - частка даних тренувань, які слід відкласти як встановлення для раннього припинення (повинно бути від 0 до 1 та використовується лише якщо true_stopping – True);

- *beta_1* - експоненціальна швидкість занепаду для оцінок вектора першого моменту в adam повинна бути в [0, 1) (використовується лише тоді, коли solver = "adam");
- *beta_2* - експоненціальна швидкість занепаду для оцінок вектора другого моменту в adam повинна бути в [0, 1) (використовується лише тоді, коли solver = "adam");
- *epsilon* - значення для чисельної стійкості в adam (використовується лише тоді, коли solver = "adam");
- *n_iter_no_change* - максимальна кількість епох, яка не відповідає поліпшенню (діє лише тоді, коли solver = 'sgd' або 'adam');
- *max_fun* - максимальна кількість викликів функцій (кількість викликів функцій буде більшим або рівним кількості ітерацій MLPRegressor).

Налаштування моделі ***AdaBoostRegressor*** містить такі параметри:

- *base_estimator* – базовий оцінювач, з якого побудований підсилений ансамбль. Якщо "Ні", то базовим оцінником є DecisionTreeRegressor (max_depth = 3);
- *n_estimators* – (default=50) – аксимальна кількість оцінювачів, у яких призупинення припиняється. У разі ідеального підходу процедуру навчання припиняють рано;
- *learning_rate* – коефіцієнт навчання скорочує внесок кожного регресора за допомогою навчання_rate. Існує компроміс між learning_rate та n_estimators.
- *loss* – {‘linear’, ‘square’, ‘exponential’} – додатковий параметр, що визначає функцію втрат, яку слід використовувати при оновленні ваг після кожної інтенсивної ітерації.

ДОДАТОК Д

АКТИ ВПРОВАДЖЕНЬ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОЇ РОБОТИ

ЗАТВЕРДЖУЮ

Начальник

Здолбунівського відділення

АТ «Рівнегаз»

_____ Климчук О.В.

«__» _____ 2019 р.

АКТ

впровадження результатів кандидатської дисертаційної роботи

Міщук Олександри Сергіївни

«НЕЙРОПОДІБНІ МЕТОДИ ТА ЗАСОБИ ПРОГНОЗУВАННЯ ПАРАМЕТРІВ ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ»

Комісія у складі Начальника Здолбунівського відділення АТ «Рівнегаз» Климчука Олександра Віталійовича, старшого інженера з експлуатаційної діяльності Полюховича Віктора Миколайовича, склали дійсний акт про те, що результати дисертаційної роботи «НЕЙРОПОДІБНІ МЕТОДИ ТА ЗАСОБИ ПРОГНОЗУВАННЯ ПАРАМЕТРІВ ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ», представлені на здобуття вченого ступеня кандидата технічних наук можуть бути використані в діяльності АТ «Рівнегаз» для локального моніторингу атмосферного повітря. Для оцінки можливості впровадження, на підприємство передано комплекс програм, де реалізовано:

- метод багатокрокового прогнозування з розширеним горизонтом прогнозування високої точності на основі корекції за допомогою комітету нейроподібних структур моделі послідовних геометричних перетворень;
- метод побудови нейроподібних структур для підвищення точності заповнення пропусків у даних за допомогою попереднього виділення компактних множин точок;
- метод швидкого заповнення пропусків, що базується на застосуванні комітету лінійних нейроподібних структур моделі послідовних геометричних перетворень і побудови на їх основі матриці коефіцієнтів лінійних поліномів.

Передбачається можливість використання результатів дисертаційної роботи при аналізі виміряних забруднюючих речовин в осередках газових установок та під час обґрунтування рекомендацій щодо прийняття управлінських рішень для уникнення надзвичайних ситуацій.

Міщук О. С. підтверджує, що передача вказаних вище матеріалів не породжує для АТ «Рівнегаз» жодних фінансових чи будь-яких інших зобов'язань ні щодо неї особисто, ні щодо третіх осіб.

Начальник Здолбунівського відділення АТ «Рівнегаз» Климчук О.В.

Старший інженер з експлуатаційної діяльності Полюхович В.М.

Здолбунівського відділення АТ «Рівнегаз»

Дисертантка Міщук О.С.

ЗАТВЕРДЖУЮ
Проректор з наукової роботи
Національного університету
«Львівська політехніка»
д.т.н., доц. _____ Демидов І. В.
«12» 12 2019 р.

АКТ
про використання результатів дисертаційної роботи Міщук Олександри Сергіївни «Нейроподібні методи та засоби прогнозування параметрів забруднення атмосферного повітря», представленої на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.23 – Системи та засоби штучного інтелекту при виконанні науково-дослідної роботи «Нейромережева технологія захисту та передачі даних у реальному часі з використанням шумоподібних кодів»

Комісія у складі голови, – начальника НДЧ, к.т.н., Небесного Р. В., та членів: завідувача кафедри інформаційних технологій видавничої справи – Ткаченка Р. О. завідувача відділу науково-організаційного супроводу наукових досліджень,– к.т.н., Лазько Г. В., та заступника начальника планово-фінансового відділу, – Чулой Т. М., цим актом підтверджують, що результати дисертаційної роботи аспірантки кафедри інформаційних технологій видавничої справи, Міщук Олександри Сергіївни, використано під час виконання науково-дослідної роботи, що фінансувалася за кошти держбюджету, за темою: «Нейромережева технологія захисту та передачі даних у реальному часі з використанням шумоподібних кодів» (номер державної реєстрації: 0119U002256).

Зокрема, Міщук О. С. розроблено метод корекції похибки багатокрокового прогнозування в реальному часі на основі комітету нейроподібних структур різних типів, що забезпечує збільшення горизонту прогнозування за рахунок зниження похибки прогнозування.

Голова комісії,
начальник НДЧ,
к.т.н. Небесний Р. В.

Члени комісії:
зав. каф. ІТВС,
д.т.н., проф. Ткаченко Р. О.

зав. відділу НОСНД,
к.т.н. Лазько Г. В.

заст. нач. ПФВ Чулой Т. М.

керівник НДР за кошти ДБ,
д.т.н., проф. Цмоць І. Г.

ЗАТВЕРДЖУЮ
Проректор з наукової роботи
Національного університету
«Львівська політехніка»
д.т.н., доц. _____ Демидов І. В.
«___» 12 2019 р.

АКТ
про використання результатів дисертаційної роботи Міщук Олександри Сергіївни «Нейроподібні методи та засоби прогнозування параметрів забруднення атмосферного повітря», представленої на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.23 – Системи та засоби штучного інтелекту при виконанні науково-дослідної роботи «Нейромережева технологія захисту та передачі даних у реальному часі з використанням шумоподібних кодів»

Комісія у складі голови, – начальника НДЧ, к.т.н., Небесного Р. В., та членів: завідувача кафедри інформаційних технологій видавничої справи – Ткаченка Р. О. завідувача відділу науково-організаційного супроводу наукових досліджень,– к.т.н., Лазько Г. В., та заступника начальника планово-фінансового відділу, – Чулой Т. М., цим актом підтверджують, що результати дисертаційної роботи аспірантки кафедри інформаційних технологій видавничої справи, Міщук Олександри Сергіївни, використано під час виконання науково-дослідної роботи, що фінансувалася за кошти держбюджету, за темою: «Нейромережева технологія захисту та передачі даних у реальному часі з використанням шумоподібних кодів» (номер державної реєстрації: 0119U002256).

Зокрема, Міщук О. С. розроблено метод корекції похибки багатокрокового прогнозування в реальному часі на основі комітету нейроподібних структур різних типів, що забезпечує збільшення горизонту прогнозування за рахунок зниження похибки прогнозування.

Голова комісії,
начальник НДЧ,
к.т.н. Небесний Р. В.

Члени комісії:
зав. каф. ІТВС,
д.т.н., проф. Ткаченко Р. О.

зав. відділу НОСНД,
к.т.н. Лазько Г. В.

заст. нач. ПФВ Чулой Т. М.

керівник НДР за кошти ДБ,
д.т.н., проф. Цмоць І. Г.